Machine Learning (2025) 114:84
https://doi.org/10.1007/s10994-024-06693-x

Re-assessing accuracy degradation: a framework for understanding DNN behavior on similar-but-non-identical test datasets

Esla Timothy Anzaku[1,2] · Haohan Wang[3] · Ajiboye Babalola[4] · Arnout Van Messem[5] · Wesley De Neve[1,2]

Received: 29 May 2024 / Revised: 12 August 2024 / Accepted: 12 December 2024

Abstract
Deep Neural Networks (DNNs) often demonstrate remarkable performance when evaluated on the test dataset used during model creation. However, their ability to generalize effectively when deployed is crucial, especially in critical applications. One approach to assess the generalization capability of a DNN model is to evaluate its performance on replicated test datasets, which are created by closely following the same methodology and procedures used to generate the original test dataset. Our investigation focuses on the performance degradation of pre-trained DNN models in multi-class classification tasks when evaluated on these replicated datasets; this performance degradation has not been entirely explained by generalization shortcomings or dataset disparities. To address this, we introduce a new evaluation framework that leverages uncertainty estimates generated by the models studied. This framework is designed to isolate the impact of variations in the evaluated test datasets and assess DNNs based on the consistency of their confidence in their predictions. By employing this framework, we can determine whether an observed performance drop is primarily caused by model inadequacy or other factors. We applied our framework to analyze 564 pre-trained DNN models across the CIFAR-10 and ImageNet benchmarks, along with their replicated versions. Contrary to common assumptions about model inadequacy, our results indicate a substantial reduction in the performance gap between the original and replicated datasets when accounting for model uncertainty. This suggests a previously unrecognized adaptability of models to minor dataset variations. Our findings emphasize the importance of understanding dataset intricacies and adopting more nuanced evaluation methods when assessing DNN model performance. This research contributes to the development of more robust and reliable DNN models, especially in critical applications where generalization performance is of utmost importance. The code to reproduce our experiments will be available at https://github.com/esla/Reassessing_DNN_Accuracy.

Keywords DNN model performance degradation · ImageNet benchmarking · ML datasets and benchmarks · Model evaluation · Multi-class classification · Pattern recognition

Editors: Kee-Eung Kim, Shou-De Lin.

Extended author information available on the last page of the article

1 Introduction

The purpose of evaluating trained machine learning models on held-out test datasets is to obtain unbiased estimates of their performance on datapoints that were not used for training purposes. A newer assessment approach in computer vision is to evaluate models on independent test datasets (Recht et al., 2019; Lu et al., 2020) that have been created by carefully replicating the way older and highly popular datasets were constructed. We expect the accuracy on the replicated datasets to be similar to the accuracy on the original datasets. However, in reality, models suffer from a substantial and consistent degradation in accuracy when evaluation is done using replicated datasets. For example, Recht et al. (2019) created two test datasets, namely CIFAR-10.1 and ImageNetV2, by carefully following the way the CIFAR-10 (Krizhevsky, 2009) and ImageNet (Deng et al., 2009) datasets were constructed, respectively. They subsequently evaluated a large number of models on these test datasets and reported substantial and consistent accuracy drops of $3-15\%$ on CIFAR-10.1 and $11-14\%$ on ImageNetV2. For convenience, in the remainder of this paper, we will refer to the original validation set of the ImageNet dataset as ImageNetV1.

Even though care is taken to ensure that the replicated datasets closely match their original datasets by design, it is reasonable to expect some characteristics of these datasets to still differ. Datasets can differ in their characteristics due to various factors, such as the under-representation of sub-populations, differences in sampling strategies (Torralba & Efros, 2011), variations in the ratio of easy to difficult datapoints (Recht et al., 2019; Lu et al., 2020), varying degrees of label errors (Northcutt et al., 2021), and statistical bias during replication (Engstrom et al., 2020). We argue that the degradation in the accuracy of a model on similar-but-non-identical test datasets does not necessarily indicate that the model fails to generalize well on the dataset. Instead, there is a possibility that the adopted evaluation approaches do not adequately capture the rich information that DNN multi-class classification models generate.

DNN multi-class classification models generally produce a softmax vector. From this vector, we can extract two outputs for any given input datapoint: the predicted label and its corresponding probability.[1] Recognizing that an accuracy comparison using all datapoints might obscure model performance nuances, we propose a method that divides datasets into subsets. These subsets are matched based on the confidence of the model studied in its predictions. This division is designed to facilitate a comparison that isolates model performance from dataset variability. When evaluating two test datasets, the division process yields only a pair of matched subsets if the datasets have the same size and their datapoint characteristics, such as class distribution and feature values, are highly similar. However, when the test datasets vary in size or exhibit differences in their datapoint characteristics, the division process inherently generates both matched and unmatched subsets. Our proposed framework utilizes all the generated subsets, both matched and unmatched, to provide a comprehensive view of model behavior across diverse dataset scenarios.

Figure 1 illustrates the central component of our framework that generates the matched and unmatched subsets. Our framework is aimed at investigating the performance degradation of a pre-trained DNN when evaluated on two test datasets – the target and source – that are expected to be similar but distinct. For instance, it can be used

[1] In this paper, we refer to the maximum value of the softmax vector as the predicted probability or confidence in an interchangeable manner.

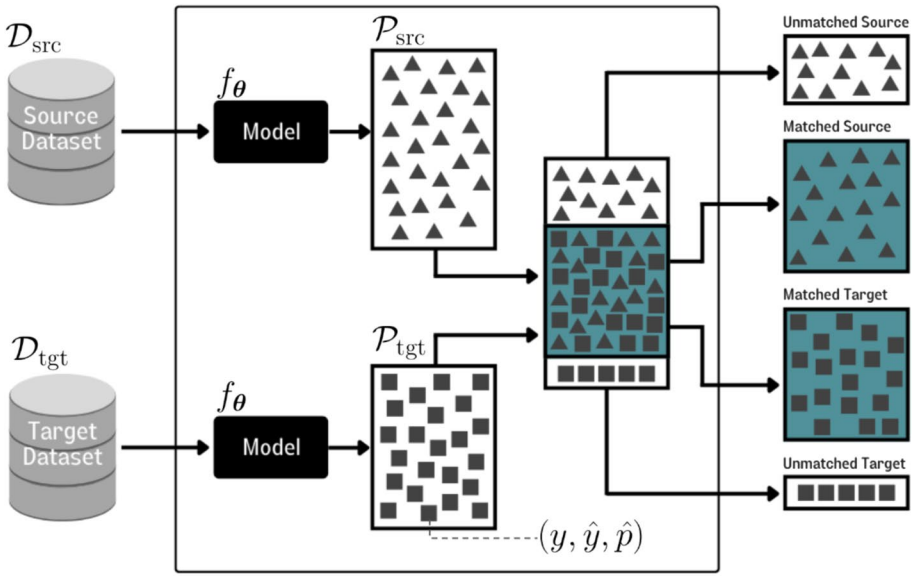


Fig. 1 Visual overview of the matching strategy in our proposed evaluation framework. Datapoints from source and target datasets are paired based on model confidence, creating matched and unmatched subsets for analysis. Details are provided in Algorithm 1

to investigate when a model performs significantly better on the source compared to the target dataset contrary to our expectations. Firstly, the predictions and confidence values for all datapoints in the test datasets are obtained from the model under evaluation. Subsequently, datapoints from the source and target datasets are matched based on predefined criteria, resulting in four subsets of matched and unmatched datapoints. These subsets are further analyzed, focusing on the accuracy of the matched subsets and the relationship between accuracy and confidence values across all subsets. Matching is performed based on two criteria: (i) both predicted labels and their corresponding confidence values, or (ii) solely the confidence values. This approach leverages these matched and unmatched subsets to identify whether differences in accuracy between the source and target datasets are due to model inadequacies. If matched subsets exhibit minimal accuracy gaps and maintain a consistent accuracy-confidence relationship, it suggests that performance discrepancies are more likely due to dataset-specific characteristics rather than model flaws. This insight is critical for further investigation into the underlying causes of performance variation.

Although it has been reported that the softmax probabilities generated by DNN models are usually over-confident (Guo et al., 2017; Hein et al., 2019), there are research efforts (Hendrycks & Gimpel, 2017; Mukhoti et al., 2020; Pearce et al., 2021) that acknowledge that softmax probabilities can serve as a baseline for the uncertainty estimates that models make about their predictions. We adopt the softmax values as the baseline for our framework, acknowledging that quantities that better capture model uncertainty could even fit better into our framework. Nevertheless, we argue that these softmax probabilities can provide additional insight into model behavior that the use of zero–one loss or accuracy alone cannot capture; one such aspect is the relationship between the accuracy and the uncertainty of a model. A property of DNN models that is

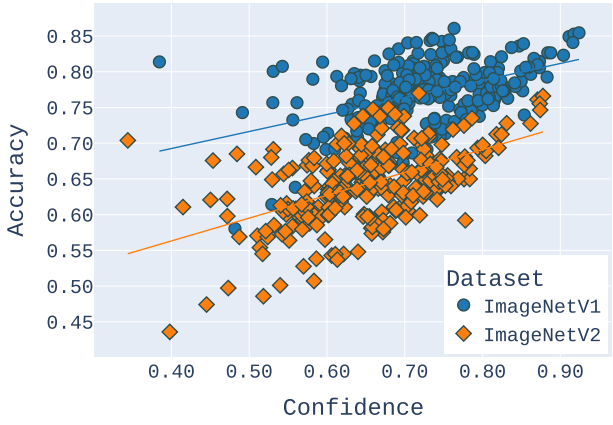


Fig. 2 Scatter plot of accuracy vs. confidence for 286 pre-trained ImageNet models, with each point representing the performance of a model on either ImageNetV1 or ImageNetV2. Confidence is the average predicted probability across all data points

of high interest to practitioners is for a DNN model to be more correct when more certain and less correct when less certain.

To observe the relationship between accuracy and confidence, we evaluated the accuracy and confidence of 286 pre-trained ImageNet models on the ImageNetV1 and ImageNetV2 datasets. Figure 2 summarizes the obtained results using a scatter plot. Here, the confidence of a model is calculated as the average of the predicted probabilities associated with the predicted labels, for all the datapoints in the test dataset of interest. In Fig. 2, we can observe that models generally tend to be less confident and less accurate on the ImageNetV2 dataset, whereas models tend to be more confident and more accurate on ImageNetV1.

The aforementioned observation brings up interesting questions that warrant further investigation. *Are the models signaling that the characteristics of the two datasets differ in more ways than we can adequately explain? Should we expect the accuracy to degrade on ImageNetV2, if indeed the characteristics of ImageNetV1 and ImageNetV2 differ but only mildly? If we find similar datapoint subsets from each of the datasets, should we expect their accuracy gap to be narrower?* Using evaluation methods that adequately capture the behavior of DNN models is a step forward toward finding answers to these kinds of questions. Our evaluation framework implicitly assesses the consistency between the accuracy metric and the predictive uncertainty across dataset subsets. For example, if we sample subsets of predictions with similar predicted probabilities (matched by predicted probabilities) from two test datasets, how large will the accuracy gap between these subsets be? Intuitively, we expect the accuracy gap to be narrow (*i*) if the uncertainty information generated by the models correlates well with the accuracy metric, and (*ii*) if the uncertainty information is consistent across the evaluated test datasets. Furthermore, for all the test splits (both matched and unmatched), we would expect to observe a similar relationship between accuracy and uncertainty estimates, even with substantial variance in accuracy across different data subsets. In Sect. 4, we present the results of our extensive evaluations conducted on 278 CIFAR-10 models and 286 ImageNet models. These results support our previously mentioned expectations and provide evidence for the effectiveness of our evaluation approach. By leveraging the uncertainty-related information generated by the models, we observe that the accuracy degradation across all evaluated models is not as severe as has been reported in prior studies.

The contributions of this paper are as follows:

1. We highlight a weakness in comparing model accuracy on similar-but-non-identical datasets; accuracy degradation on replicate datasets is usually accompanied by higher model predictive uncertainty, which is often not taken into account.
2. We propose a nuanced evaluation framework that considers both predicted labels and their corresponding probabilities, enabling a deeper understanding of model behavior across multiple test datasets.
3. Through an extensive evaluation of 564 pre-trained DNN models, we provide empirical evidence that challenges the reported deterioration in accuracy on similar-but-non-identical datasets. By effectively utilizing the uncertainty-related information produced by these models, we infer that the performance of the models does not severely degrade on the evaluated replicated datasets, suggesting that other dataset-related factors could contribute to the observed degradation.

In the remainder of this paper, we describe our evaluation approach in detail in Sect. 2. Furthermore, we summarize related research efforts in Sect. 3. Next, we discuss our experimental setup and our experimental results in Sect. 4. Finally, we present our conclusions and a of number directions for future research in Sect. 5.

2 Proposed evaluation framework

2.1 Notations

In this section, we describe our evaluation framework, which enables us to gain more insight into the performance of models when evaluated using multiple datasets. Although the framework could be used to evaluate any supervised classification model that generates predicted labels and their corresponding probabilities, we restrict our scope to supervised DNN classification models for computer vision in this paper. Supervised classification in machine learning involves observing examples of a random vector $\boldsymbol{X}$ and an associated random variable Y to learn to predict Y from $\boldsymbol{X}$ by estimating the conditional probability distribution $P(Y|\boldsymbol{X})$. Our evaluation framework assumes that we are provided with:

- an *i.i.d.* dataset $\mathcal{D} = \left\{\left(\boldsymbol{x}^{(i)}, y^{(i)}\right)\right\}_{i=1}^{N} \subset \mathcal{X} \times \mathcal{Y}$, which is further partitioned into disjoint subsets: train ($\mathcal{D}_{\text{train}}$), validation ($\mathcal{D}_{\text{val}}$), and test ($\mathcal{D}_{\text{test1}}$)[2]. Here, $\mathcal{X}$ and $\mathcal{Y} = \{1, 2, ..., K\}$ denote the input space and label space, respectively, and K represents the number of classes, labels, or categories that each realization of $\boldsymbol{X}$ can be assigned to. An example input space may comprise a set of raw images belonging to the categories (denoted by integers) in a given label space;
- a classification model $f_{\boldsymbol{\theta}} : \mathcal{X} \rightarrow \mathcal{Y}$ with trained parameters $\boldsymbol{\theta}$. This classification model, which has been trained on $\mathcal{D}_{\text{train}}$ and validated on $\mathcal{D}_{\text{val}}$, produces the predicted class label $\hat{y} = \arg\max_{y \in \mathcal{Y}} P(Y = y|\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{\theta})$ and the corresponding predicted probability $\hat{p} = \max_{y \in \mathcal{Y}} P(Y = y|\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{\theta})$; and

[2] The ImageNet dataset does not include the ground truth labels for the test partition. Therefore, the accuracy on the validation partition is usually reported in published articles.

- a second test dataset, $\mathcal{D}_{\text{test2}} = \{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{M} \subset \mathcal{X} \times \mathcal{Y}$, such that $\mathcal{D} \cap \mathcal{D}_{\text{test2}} = \emptyset$.

Algorithm 1 Strategies for matching two datasets: based on both predicted labels and probabilities (bottom left, referred to as MATCH1), and based on probabilities only (bottom right, referred to as MATCH2).

procedure MATCHPREDICTIONS($f_{\boldsymbol{\theta}}$, $\mathcal{D}_{\text{src}}$, $\mathcal{D}_{\text{tgt}}$, ϵ)
  $\mathcal{P}_{\text{src}} \Leftarrow$ PREDICT($f_{\boldsymbol{\theta}}, \mathcal{D}_{\text{src}}$)
  $\mathcal{P}_{\text{tgt}} \Leftarrow$ PREDICT($f_{\boldsymbol{\theta}}, \mathcal{D}_{\text{tgt}}$)
  Initialize $\mathcal{P}_{\text{src\_matched}}, \mathcal{P}_{\text{tgt\_matched}}, \mathcal{P}_{\text{tgt\_unmatched}}$ as empty
  $N \Leftarrow$ LENGTH($\mathcal{P}_{\text{tgt}}$)
  for $i \Leftarrow 1, N$ **do**
    $y_t, \hat{y}_t, \hat{p}_t \Leftarrow \mathcal{P}_{\text{tgt}}[i]$
    $y_s, \hat{y}_s, \hat{p}_s, is\_match \Leftarrow$ MATCH1(...) or MATCH2(...)
    if is_match **then**
      Add matched items to $\mathcal{P}_{\text{src\_matched}}$ and $\mathcal{P}_{\text{tgt\_matched}}$
    else
      Add to $\mathcal{P}_{\text{tgt\_unmatched}}$
    end if
  end for
  $\mathcal{P}_{\text{src\_unmatched}} = \mathcal{P}_{\text{src}} - \mathcal{P}_{\text{src\_matched}}$
  return Matched and unmatched sets
end procedure

function MATCH1(y_t, $\hat{y}_t$, $\hat{p}_t$, $\mathcal{P}_{\text{src}}$, ϵ)
  Initialize $\mathcal{P}_{\text{matched}}$ as empty
  for all $(y, \hat{y}, \hat{p}) \in \mathcal{P}_{\text{src}}$ **do**
    if $\hat{y}_t = \hat{y}$ and $\hat{p}_t \in [\hat{p} - \epsilon, \hat{p} + \epsilon]$ **then**
      Add $(y, \hat{y}, \hat{p})$ to $\mathcal{P}_{\text{matched}}$
    end if
  end for
  if $|\mathcal{P}_{\text{matched}}| > 0$ **then**
    $(y_s, \hat{y}_s, \hat{p}_s) \Leftarrow$ RANSEL($\mathcal{P}_{\text{matched}}$)
    Remove $(y, \hat{y}, \hat{p})$ from $\mathcal{P}_{\text{src}}$
    return $(y_s, \hat{y}_s, \hat{p}_s, \text{True})$
  else
    return (null, null, null, False)
  end if
end function

function MATCH2(y_t, $\hat{y}_t$, $\hat{p}_t$, $\mathcal{P}_{\text{src}}$, ϵ)
  Initialize $\mathcal{P}_{\text{matched}}$ as empty
  for all $(y, \hat{y}, \hat{p}) \in \mathcal{P}_{\text{src}}$ **do**
    if $\hat{p}_t \in [\hat{p} - \epsilon, \hat{p} + \epsilon]$ **then**
      Add $(y, \hat{y}, \hat{p})$ to $\mathcal{P}_{\text{matched}}$
    end if
  end for
  if $|\mathcal{P}_{\text{matched}}| > 0$ **then**
    $(y_s, \hat{y}_s, \hat{p}_s) \Leftarrow$ RANSEL($\mathcal{P}_{\text{matched}}$)
    Remove matched item from $\mathcal{P}_{\text{src}}$
    return $(y_s, \hat{y}_s, \hat{p}_s, \text{True})$
  else
    return (null, null, null, False)
  end if
end function

Comments:

- The PREDICT(.) function returns the predictions of a model in an array that consists of tuples $(y, \hat{y}, \hat{p})$, where y, $\hat{y}$, and $\hat{p}$ denote the ground truth label, the predicted label, and the predicted probability, respectively.
- The RANSEL(.) function returns a randomly selected tuple $(y, \hat{y}, \hat{p})$ from the array that contains the input tuples.

The task at hand is to compare the effectiveness of $f_{\boldsymbol{\theta}}$ on $\mathcal{D}_{\text{test1}}$ and $\mathcal{D}_{\text{test2}}$ using certain metrics, such as accuracy.

In Sect. 1, we introduced two matching criteria: (*i*) matching by both the predicted labels and their corresponding predicted probabilities and (*ii*) matching by the predicted probabilities only. In what follows, we describe our proposed evaluation framework. Given two test datasets $\mathcal{D}_{\text{test1}}$ and $\mathcal{D}_{\text{test2}}$, our evaluation strategy aims at assessing the accuracy of a model on these datasets, while taking into account the predicted probabilities generated by this model. For convenience, we designate the dataset with the larger number of datapoints as the source dataset $\mathcal{D}_{\text{src}}$, and the other dataset as the target dataset $\mathcal{D}_{\text{tgt}}$.

First, the model of interest is used to generate the predicted labels and probabilities for all the datapoints in both test datasets. We represent the output of the model, for a datapoint, as a tuple $(y, \hat{y}, \hat{p})$,[3] where y, $\hat{y}$, and $\hat{p}$ represent the ground truth label, the predicted label, and the predicted probability, respectively. In a next step, we generate matched and unmatched subsets from $\mathcal{P}_{\text{src}}$ and $\mathcal{P}_{\text{tgt}}$, whose elements are tuples $(y, \hat{y}, \hat{p})$, by sub-sampling from all the model outputs obtained for the datasets $\mathcal{D}_{\text{src}}$ and $\mathcal{D}_{\text{tgt}}$ without replacement, making use of one of the matching criteria. Our first matching criterion is summarized as follows: for every element in $\mathcal{P}_{\text{tgt}}$, find an element in $\mathcal{P}_{\text{src}}$, such that they both have the same predicted labels and such that their predicted probabilities are approximately equal. In this context, we make use of a very small fraction ϵ to control how approximately matched the probability values should be. For example, if we want the probability difference to be within $\pm 1\%$, we set $\epsilon = 0.01$. The second matching criterion relaxes the need for the predicted labels to be the same and matches only the predicted probabilities. Test datasets $\mathcal{D}_{\text{src}}$ and $\mathcal{D}_{\text{tgt}}$ could have different number of datapoints. Moreover, some datapoints could have probabilities that are not matched. Therefore, the matching strategy could result in four non-empty sets: two matched and two unmatched sets. Our evaluation framework, using these matching criteria, is described in more detail in Algorithm 1 and a summarizing overview is presented in Fig. 1.

2.2 Accuracy evaluation

Algorithm 1 generates four output arrays: $\mathcal{P}_{\text{src\_matched}}$, $\mathcal{P}_{\text{tgt\_matched}}$, $\mathcal{P}_{\text{src\_unmatched}}$, and $\mathcal{P}_{\text{tgt\_unmatched}}$. The arrays $\mathcal{P}_{\text{src\_matched}}$ and $\mathcal{P}_{\text{tgt\_matched}}$ contain information about the model outputs that satisfy the predefined matching criteria. These criteria determine whether a datapoint from the source dataset $\mathcal{D}_{\text{src}}$ has a corresponding match in the target dataset $\mathcal{D}_{\text{tgt}}$. Conversely, the arrays $\mathcal{P}_{\text{src\_unmatched}}$ and $\mathcal{P}_{\text{tgt\_unmatched}}$ store information about the model outputs for datapoints that do not have matching counterparts in the other dataset according to the same criteria. Specifically, $\mathcal{P}_{\text{src\_unmatched}}$ contains information for datapoints in $\mathcal{D}_{\text{src}}$ that do not have a match in $\mathcal{D}_{\text{tgt}}$, while $\mathcal{P}_{\text{tgt\_unmatched}}$ contains information for datapoints in $\mathcal{D}_{\text{tgt}}$ that do not have a match in $\mathcal{D}_{\text{src}}$.

Although many other metrics can be used to evaluate the aforementioned arrays to get a better understanding of the datapoints that are considered similar or not similar, as judged by the outputs of a model, the scope of this paper is restricted to the accuracy metric, given that this metric is widely used for evaluating models on multiple test datasets. Our

[3] Even though the $\hat{y}$ and $\hat{p}$ are sufficient for the generation of matched subsets, since we need to calculate the accuracy on the matched subsets, we adopt a tuple representation for the model output obtained for each datapoint, with this representation including the ground truth label.

definition of accuracy is conventional – the ratio of the number of correctly classified datapoints to the total number of datapoints under evaluation. However, we make a distinction between accuracy evaluated on all the datapoints in a source or target dataset, and the accuracy evaluated on the matched or unmatched datapoints. We refer to the accuracy on all datapoints simply as *accuracy*, the accuracy on the matched subsets as *matched accuracy*, and the accuracy on the unmatched subsets as *unmatched accuracy*.

3 Related research

In this section, we briefly summarize research efforts that are related to our work.

Possible explanations for the accuracy degradation in replication datasets Even after carefully following the way the original datasets were constructed, the creators of the CIFAR-10.1 and the ImageNetV2 datasets (Recht et al., 2019) concluded that the degradation in accuracy is not resulting from adaptive over-fitting to the original test datasets, suggesting that a possible explanation for the degradation is the presence of harder-to-classify images in the newer test datasets. Engstrom et al. (2020) concluded that the accuracy degradation in the ImageNetV2 dataset is the result of statistical bias in the creation of the ImageNetV2 dataset. They show that the accuracy gap is indeed narrower if various sources of statistical bias are accounted for. Specifically, they demonstrate that only $3.6\% \pm 1.5\%$ of the accuracy drop for ImageNetV2 remains unaccounted for. Our experimental results support their finding and our proposed evaluation framework additionally provides a practical tool for realistic performance assessment of DNN models on multiple test datasets.

The versatility of softmax output in DNN research Softmax output has become a fundamental baseline across various research domains within DNN research, demonstrating its broad applicability and adaptability.

In the domain of uncertainty estimation and out-of-distribution (OOD) detection, softmax output has been established as a baseline for identifying misclassified and OOD samples (Hendrycks & Gimpel, 2017). Researchers have acknowledged its potential for capturing prediction uncertainty in multi-class classification tasks (Mukhoti et al., 2020; Pearce et al., 2021) and have developed techniques to enhance OOD detection capabilities by building upon this baseline (Liang et al., 2018). However, a recent study found that current benchmark studies for evaluating OOD detectors are highly sensitive to experimental details such as random seed, train/val/test splits, and early stopping (Szyc et al., 2023). The study reported that the highest instability was observed for OOD detectors such as the k-nearest neighbors (KNN) and Mahalanobis (Lee et al., 2018), compared to logit-based methods like Maximum Softmax Probability (MSP) (Hendrycks & Gimpel, 2017).

Softmax output plays a crucial role in various learning paradigms, including semi-supervised learning, self-supervised learning, and knowledge distillation. In semi-supervised learning, particularly with pseudo-labeling strategies (Lee, 2013; Sohn et al., 2020), the highest confidence predictions of a model are used as 'pseudo-labels' to extend training to unlabeled data. In self-supervised learning methods such as DINO (self-distillation with no labels) (Caron et al., 2021), the softmax output of a teacher network serves as the target for a student network, enabling the learning of meaningful representations without labeled data. Similarly, in knowledge distillation (Hinton et al., 2015), softened softmax probabilities are utilized as teaching signals to transfer knowledge from complex models to simpler, computationally efficient ones. The softened probabilities provide a richer form of

guidance than hard labels, allowing student models to learn subtle decision boundaries and inter-class relationships (Hinton et al., 2015; Romero et al., 2015).

Northcutt et al. (2021) utilized softmax probabilities within their Confident Learning framework to identify and rectify label errors in training and test datasets. Similarly, Sun and Lampert (2020) employed softmax outputs in their KS(conf) approach to detect dataset shifts, which is crucial for assessing the reliability of DNN models in real-world scenarios. Stephan Rabanser et al. (2019) also leveraged statistical tests on softmax outputs in their Black Box Shift Detection (BBSD) method to identify potential out-of-distribution data and reveal subtle shifts in data distribution.

Despite its utility, the interpretation of softmax output as a definitive measure of uncertainty should be approached with caution. Modern neural networks have been found to be poorly calibrated, with various factors influencing calibration (Guo et al., 2017). Moreover, ReLU-type neural networks can exhibit arbitrarily high confidence far away from the training data, with the problem exacerbated by increasing network depth (Hein et al., 2019). These findings highlight the propensity of softmax outputs to exhibit overconfidence, which can result in misleading uncertainty estimates.

Notwithstanding these limitations, we posit that softmax outputs can still serve as a valuable baseline for investigating the behavior of DNN models within the specific context and experimental settings of this paper. Our work leverages softmax output to investigate the robustness and generalization performance of DNN models on unseen test datasets that are similar to the training data, and for which we do not expect substantial performance degradation. Unlike previous works that utilized softmax for tasks such as detecting misclassifications, semi-supervised learning, knowledge distillation, dataset shift detection, and OOD detection, our focus is on employing softmax as a tool for investigating model behavior and performance inconsistency under test scenarios where performance degradation is not expected.

Calibration of models Model calibration, also known as confidence calibration, is the task of predicting probability estimates that accurately reflect the true likelihood of correctness (Guo et al., 2017). Existing methods for model calibration can be categorized into two main approaches: (*i*) improving the modeling process to create inherently calibrated models (Kumar et al., 2018; Mukhoti et al., 2021; Krishnan & Tickoo, 2020), and (*ii*) applying post-processing techniques to enhance the outputs of pre-trained models (Naeini et al., 2015; Guo et al., 2017).

Expected Calibration Error (ECE) (Naeini et al., 2015; Guo et al., 2017) is a widely used metric for assessing model calibration, providing a single scalar value that summarizes the calibration performance of a model across different predicted probability intervals. Despite its popularity, ECE has several limitations. First, as a histogram-based metric, ECE is heavily influenced by the chosen binning strategy (Nixon et al., 2019; Ding et al., 2019). Second, different models may generate varying distributions of predicted probabilities for the same dataset, making ECE less suitable for comparing calibration performance across models. Finally, ECE is independent of accuracy, implying that a model can be well-calibrated but have low accuracy, or vice versa, suggesting a trade-off between accuracy and calibration.

To address these limitations, we utilize an approach inspired by the work of Bröcker and Smith (2007). They introduced a resampling method for assigning consistency bars to observed frequencies in reliability diagrams, enabling visual evaluation of the likelihood of the observed relative frequencies under the assumption of reliably predicted probabilities. Inspired by their work, we plot accuracy versus confidence using histogram bins and augment the plot with confidence interval regions. This addition is

motivated by the first two limitations of ECE highlighted in the preceding paragraph. Incorporating confidence intervals provides a more informative assessment of the calibration performance of a model by accounting for the variability in sample sizes across different histogram bins.

Classification with a reject option Classifiers with a reject option can abstain from predicting a label if certain criteria are not met. Examples of classifiers with a reject option are those where the rejection is based on confidence thresholds (Chu, 1965; Cortes et al., 2016). This group of classifiers comes with a trade-off between error rates and rejection rates, benefiting from reliable confidence estimation. Research efforts on classification with a reject option focus on creating classifiers that demonstrate improved effectiveness while having the ability to abstain from making certain predictions. In contrast, our work aims to establish a systematic approach for evaluating any multi-class classification DNN model. This is particularly crucial when assessing the performance of a model on test datasets for which we expect no substantial performance degradation.

Matching methods for causal inference

Matching methods are statistical techniques designed to approximate the conditions of randomized experiments in observational studies by ensuring that treated and control groups have similar covariate distributions (Stuart, 2010). These methods, including propensity score matching and subclassification, aim to reduce bias by creating well-matched samples from the original groups, thus facilitating more accurate causal inferences. Early works on matching began in the 1940 s (Stuart, 2010), but their popularity has grown across various fields such as political science (Ho et al., 2007), economics (Chiappori and Salanié, 2016), epidemiology (Brookhart et al., 2006), medicine (Thomas et al., 2020), and statistics (Rubin, 2006) due to their effectiveness in addressing confounding variables and enabling robust analyses of observational data. Our proposed method, which involves matching model predictions between two test datasets, is inspired by these matching techniques. By matching predictions according to specific model output criteria, we aim to reduce the impact of potential unknown differences between two similar test datasets, thereby enabling a more accurate assessment of the performance of the model on both datasets.

4 Experimental setup and results

4.1 Experimental setup

Dataset pairs The goal of our experiments is to evaluate a representative sample of published DNN classification models on the CIFAR-10 and ImageNet datasets, using the evaluation framework proposed in Sect. 2. To that end, we trained 278 models on the CIFAR-10 dataset using PyTorch implementations by Mukhoti et al. (2020)[4] under various training settings. These training settings include different model architectures, loss functions, learning rate schedules, data augmentation techniques, and random seeds. We describe the datasets used and the various characteristics of the trained models in Table 1 and Table 2, respectively, both located in Appendix A. For CIFAR-10, we carried out evaluations on the following test dataset pairs: (*i*) CIFAR-10 versus CIFAR-10.1, (*ii*) CIFAR-10 versus

[4] We additionally used models implemented in https://github.com/hysts/pytorch_image_classification.

CIFAR-10.2, and (*iii*) CIFAR-10 versus CINIC-10. The CIFAR-10.2 dataset was created by Lu et al. (2020) and is a variant of the CIFAR-10 dataset; it was derived from the same source as CIFAR-10 and assembled via a similar process. The CINIC-10 dataset (Darlow et al., 2018) was derived from ImageNet, and consists of images that belong to the same classes as those of the CIFAR-10 dataset; all the images were resized to a resolution of 32×32.

Due to computational limitations, we did not re-train the ImageNet models but utilized 286 pre-trained published models from a popular PyTorch repository (Wightman, 2019). The network architectures of a selected number of the pre-trained ImageNet models and their names, as used in the source GitHub repository, are presented in Table 3 of Appendix A. Similar to what was done for the CIFAR-10 pairs, we determine the accuracy and matched accuracy for the ImageNetV1 versus ImageNetV2 dataset pair for each pre-trained model. The ImageNetV1 dataset consists of a total of 50, 000 images taken from 1, 000 categories, while ImageNetV2 consists of a total of 10, 000 images taken from 1, 000 categories.

Generated results We applied Algorithm 1 to each of the aforementioned dataset pairs to generate matching subsets, with $\epsilon = 0.01$. Subsequently, each model is evaluated on these subsets to determine (*i*) the *accuracy* on the source and the target datasets, (*ii*) the *matched accuracy* on the matched source and matched target datasets, and (*iii*) the *unmatched accuracy* on the unmatched source and target subsets. Another value calculated is the fraction of datapoints in the target dataset that does not find a match with datapoints in the source dataset, which we refer to as *Fraction Unmatched*. These results are presented in Fig. 3 and Fig. 4. Additional results can be found in Fig. 7 and of Appendix A. The accuracy and matched accuracy gaps are better visualized in the plots on the right column of these Figures.

The results for the matched accuracy are the average of 10 runs for each of the matching criteria (refer to Algorithm 1 for their description). Since the standard errors over these runs were negligible, we did not include error bars for the matched accuracy. For accuracy, we evaluate already trained models and it is not feasible to have the same pre-trained models over multiple seeds. Therefore, we plot the accuracy of the available pre-trained models.

4.2 Results and discussion

Consistent and significant accuracy drops Our extensive evaluation of the accuracy of a plethora of models, on both the original and replication datasets, produced results that are in line with the results reported by Recht et al. (2019), Miller et al. (2021), and Lu et al. (2020). When accuracy is evaluated on all datapoints, we observe consistent but significant accuracy drops on the replicated datasets. These results are shown in Figs. 3a 3c, 3e, and 4a as the gap between the solid blue line and the solid orange line; the difference in accuracy can be better viewed in the corresponding accuracy difference plots in Figs. 3b, 3d, 3f, and 4b as the blue line.

Narrower matched accuracy gaps Given an input image, a model is used to predict the label for this image and the likelihood of the correctness of the prediction, which we loosely interpret as the belief of the model about the correctness of the prediction or its uncertainty. Algorithm 1 enables us to effectively partition the evaluation datasets into matching and non-matching subsets and to evaluate how consistently the accuracy of models aligns with their uncertainty estimates.

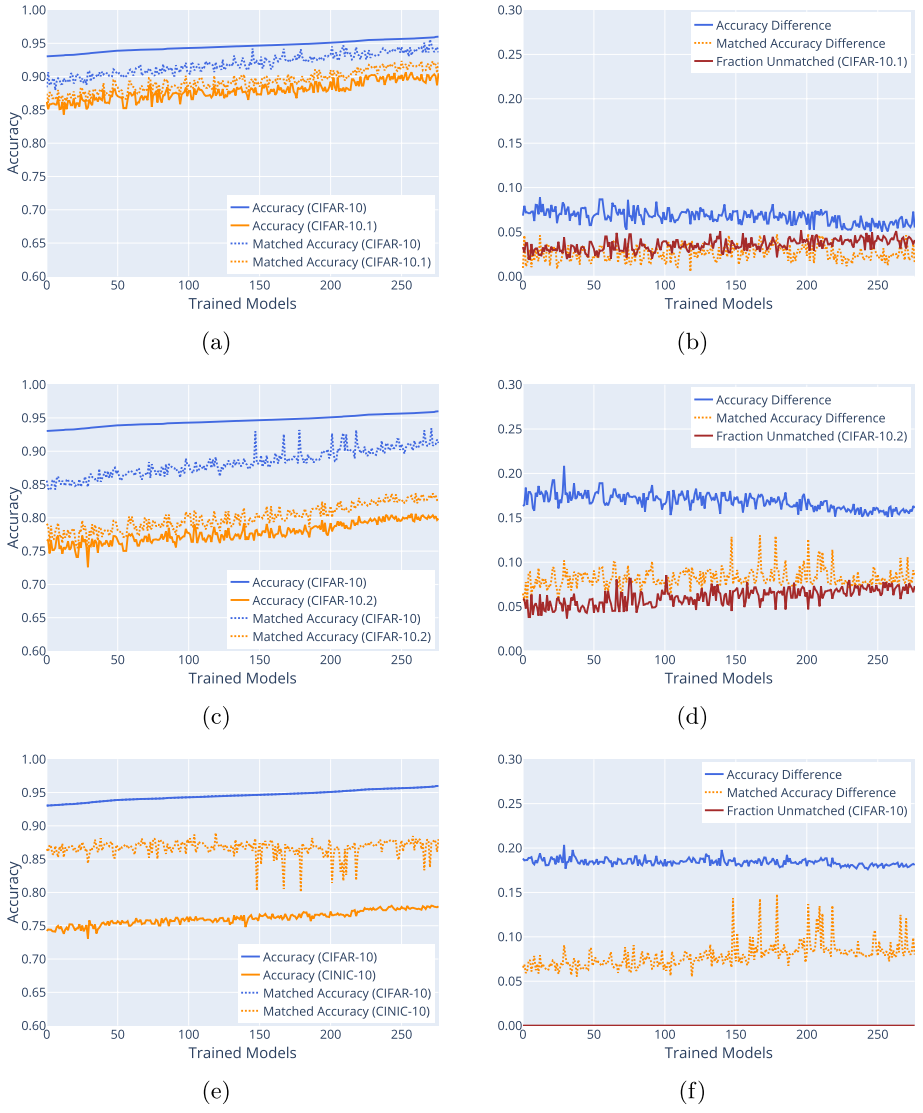


Fig. 3 Plots for the accuracy and matched accuracy for (a) CIFAR-10 versus CIFAR-10.1, (c) CIFAR-10 versus CIFAR-10.2, and (e) CIFAR-10 versus CINIC-10. Plots for accuracy difference, matched accuracy difference, and fraction unmatched for (b) CIFAR-10 versus CIFAR-10.1, (d) CIFAR-10 versus CIFAR-10.2, and (f) CIFAR-10 versus CINIC-10. The models in all plots are sorted according to increasing accuracy on CIFAR-10

By utilizing our evaluation framework, which allows taking into account uncertainty information, the matched accuracy gap is consistently narrower, indicating that the accuracies of the models on the test dataset pairs are closer if the uncertainty information is taken into account. The key takeaway message here is that the uncertainty that a model generates provides additional information that should be taken into account to better evaluate the model. These results are presented in Figs. 3a, 3c, 3e, and 4a as the gap

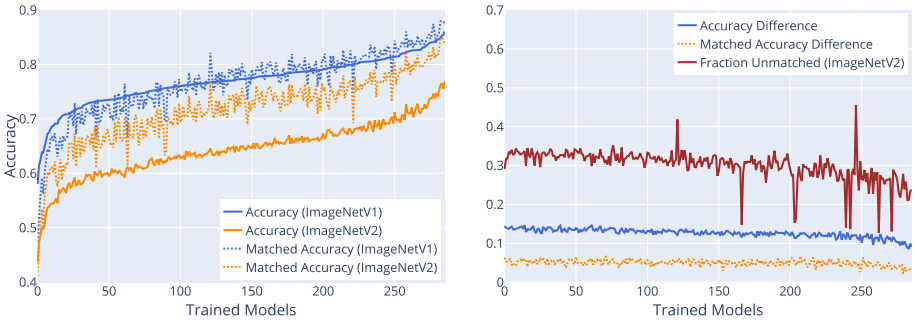


Fig. 4 (a) Accuracy and matched accuracy plots and (b) difference in accuracy and matched accuracy for 286 models evaluated on the ImageNet and ImageNetV2 dataset pair according to Algorithm 1. The models in all plots are sorted according to increasing accuracy on ImageNetV1

between the dotted blue line and the dotted orange line; the accuracy difference can better be viewed in Figures 3b, 3d, 3f, and 4b as the orange line. The evaluation results obtained when matching is done by predicted probabilities only (our second matching criterion) can be found in Fig. 7 in Appendix A. While the plots effectively illustrate the reduction in the performance gap using our evaluation framework, we also quantify this reduction by calculating the percentage decrease in the matched accuracy gap relative to the original accuracy gap. The accuracy gap decreased by 60.83% for CIFAR-10.1, 51.30% for CIFAR-10.2, 57.06% for CINIC-10, and 64.05% for ImageNetV2, on average across all models.

Consistency of confidence versus accuracy across various subsets of the test datasets

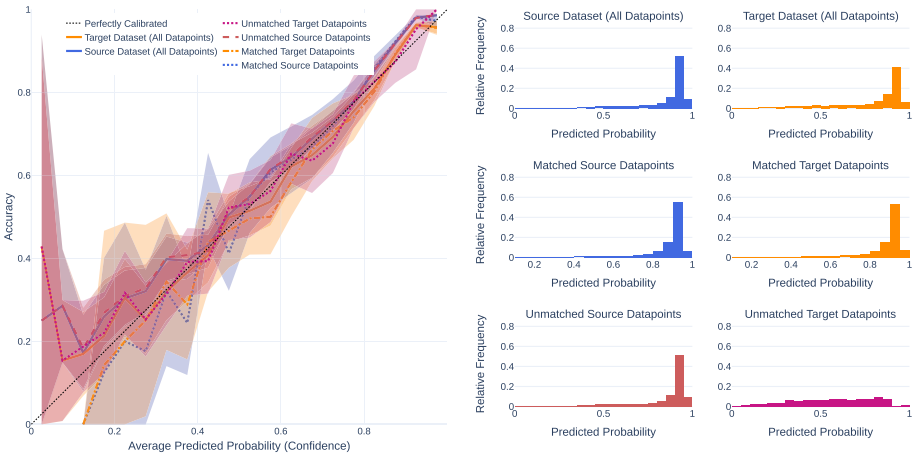
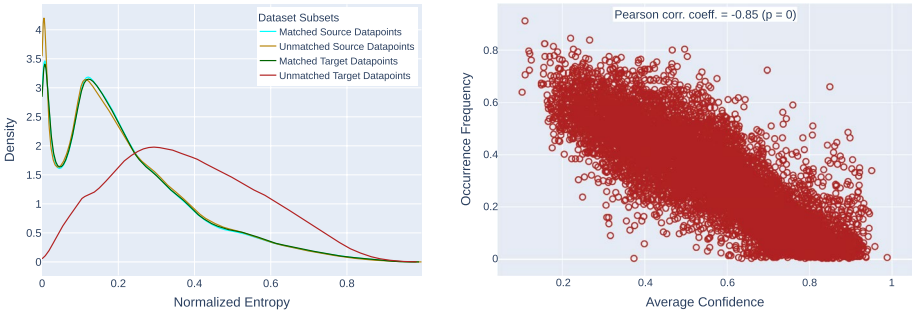


Fig. 5 The plots of characteristics of a pre-trained BEiT model (Dong et al., 2022) when evaluated on various subsets of the ImageNetV1 and ImageNetV2 datasets, as generated by our proposed evaluation framework: (a) accuracy versus confidence and (b) density plots for the predicted probabilities



(a) (b)

Fig. 6 (a) Density plots of normalized entropy for matched and unmatched subsets of ImageNetV1 (source) and ImageNetV2 (target) datasets across all evaluated models. Higher entropy values indicate greater prediction uncertainty, with a rightward shift in the density curve for unmatched target datapoints, indicating a higher proportion of uncertain predictions in this subset. (b) Scatter plot illustrating the relationship between average confidence and occurrence frequency of unmatched datapoints in the target dataset across models. A strong negative correlation (Pearson correlation coefficient $= -0.85$, $p < 0.001$) is observed, suggesting that datapoints consistently assigned lower confidence by models are more likely to appear frequently in unmatched subsets

To further illustrate the behavior of models for the various dataset subsets, we present in Fig. 5a the accuracy versus confidence curves, which are also known as calibration curves or reliability diagrams (Niculescu-Mizil & Caruana, 2005), for all the subsets of the ImageNetV1 and the ImageNetV2 datasets. The predictions were generated by utilizing our proposed framework to evaluate an ImageNet pre-trained BEiT model (Dong et al., 2022).[5] In Fig. 5b, we present the density plots for the predicted probabilities. The accuracy gap is 7.99%, while the matched accuracy gap is 2.58%. The accuracy for all datapoints in ImageNetV1 is 88.57% compared to 80.58% in ImageNetV2. For the matched subsets, the accuracies are 90.27% for ImageNetV1 and 87.69% for ImageNetV2. For the unmatched subsets, the accuracies are 88.29% for ImageNetV1 and 56.59% for ImageNetV2. These results indicate that accuracy across different test subsets could vary substantially when their characteristics differ. Even though this variation exists, the model demonstrates consistent behavior within these subsets, showing a stable correlation between the generated uncertainty values and prediction accuracy. This consistency is crucial, especially under mild distribution shifts, as it allows performance estimation from generated uncertainty values. This is valuable for applications like classification with rejection, where the model can abstain from making decisions for high-uncertainty predictions, thus enhancing the reliability of decision-making.

Explanation for the perceived accuracy gap

To investigate the causes of the perceived accuracy gap, we analyzed the uncertainty in matched and unmatched subsets across all models for both the source (ImageNetV1) and target (ImageNetV2) datasets. Uncertainty was quantified using normalized entropy, $H_{\text{norm}} = -\frac{1}{\log K} \sum_{i=1}^{K} p_i \log p_i$, where K is the number of classes and p_i is the predicted probability for class i. This normalization bounds entropy between 0 and 1, providing a clear interpretation of uncertainty levels.

[5] The pre-trained model, named as beit_large_patch16_512.in22k_ft_in22k_in1k, was obtained from https://github.com/rwightman/pytorch-image-models.

Table 1 Description of the dataset partitions for CIFAR-10 and its replication datasets

Name	Default available	Partition used	Total images
CIFAR-10	Train, Test	Test	10,000
CIFAR-10.1	Test	Test	2,000
CIFAR-10.2	Test	Test	10,000
CINIC-10	Train, Validation, Test	Test	90,000

Table 2 Description of the characteristics of the trained models for CIFAR-10

Network architectures	VGG (Simonyan & Zisserman, 2015), ResNet (He et al., 2016), DenseNet (Huang et al., 2017), ResNeXt (Xie et al., 2017), ResNet_Preact (He et al., 2016), Shake_Shake (Gastaldi, 2017), PyramidNet (Han et al., 2017), SENet (Hu et al., 2018)
Loss functions	Focal Loss (Lin et al., 2017), Brier Loss (Brier, 1950), Maximum Mean Calibration Error (MMCE) Loss (Kumar et al., 2018), Label Smoothing Loss (Müller et al., 2019)
Data augmentations (Basic)	Random Cropping, Horizontal Flipping
Data augmentations (Advanced)	CutMix (Yun et al., 2019), MixUp (Zhang et al., 2017), RICAP (Takahashi et al., 2018)

Table 3 Network architectures and names of a select number of models pre-trained on ImageNet

Network architectures	VGG (Simonyan & Zisserman, 2015), DenseNet (Huang et al., 2017), ResNet (He et al., 2016), ResNeXt (Xie et al., 2017), Inception-ResNet (Szegedy et al., 2017), EfficientNet (Tan & Le, 2019), DPN (Chen et al., 2017), SEResNet (Hu et al., 2018), HRNet (Zhang et al., 2021), DeiT (Touvron et al., 2021), PNasNet (Liu et al., 2017), Visformer (Chen et al., 2021), Vision Transformers (ViT) (Dosovitskiy et al., 2021)
Pre-trained models (Wightman, 2019)	densenet161, deit_tiny_distilled_patch16_224, dpn68b, efficientnet_b2, dpn92, gluon_resnet50_v1d, gluon_seresnext50_32x4d, gluon_seresnext101_32x4d, hrnet_w18_small_v2, ig_resnext101_32x8d, ig_resnext101_32x16d, legacy_seresnet50, ig_resnext101_32x32d, ig_resnext101_32x48d, inception_resnet_v2, pnasnet5large, resnet18, seresnext50_32x4d, tf_efficientnet_b5, tf_efficientnet_l2_ns, vgg13, visformer_small, vit_base_patch16_224, vit_tiny_patch16_384

Figure 6a presents density plots of normalized entropy values for datapoints in each subset across all evaluated models. The unmatched datapoints from the target dataset exhibit higher entropy, indicated by a rightward shift in their density curve, suggesting a prevalence of high-entropy predictions in comparison to other subsets. To further understand this observation, we calculated the occurrence frequency for each datapoint in the unmatched subsets (i.e., the proportion of models in which a datapoint was unmatched) and the average confidence score assigned by these models. The presence of datapoints that are present in all subsets of the unmatched dataset could facilitate the investigation of the specific dataset properties that could differ between the target and source datasets. However,

Fig. 7 Plots for accuracy and matched accuracy, when matched by predicted probabilities only for the following dataset pairs: (a) CIFAR-10 versus CIFAR-10.1, (c) CIFAR-10 versus CIFAR-10.2, (e) CIFAR-10 versus CINIC-10(h), and ImageNetV1 versus ImagenetV2. Plots for accuracy difference, matched accuracy difference, and fraction unmatched for (b) CIFAR-10 versus CIFAR-10.1, (d) CIFAR-10 versus CIFAR-10.2, (f) CIFAR-10 versus CINIC-10, and (h) ImageNetV1 versus ImagenetV. The models in all plots are sorted according to increasing accuracy on CIFAR-10 and ImageNetV1

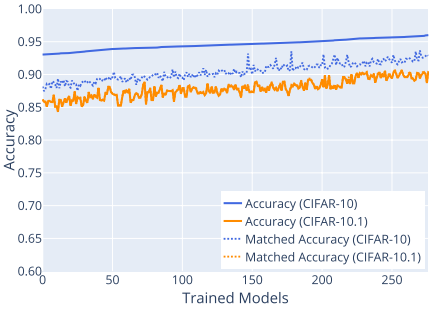
we expect some variability in the composition of the subsets due to the factors discussed in Appendix B.

Figure 6b illustrates the relationship between the occurrence frequency and average confidence, revealing a strong negative correlation (Pearson coefficient $= -0.85$, $p < 0.001$). This result indicates that datapoints consistently assigned lower confidence across models are more likely to appear frequently in the unmatched sets. These findings, combined with our earlier observations, suggest that the perceived accuracy degradation may stem from dataset-specific properties that induce greater model uncertainty in the target dataset. This observation warrants further investigation into the properties of the datasets and potential issues with existing modeling and evaluation practices.

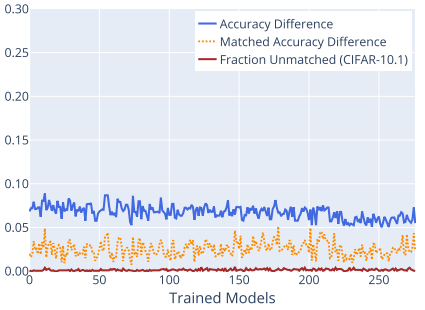
5 Conclusions and future work

In this paper, we introduced a new evaluation framework that addresses the limitations of conventional accuracy-based methods when comparing the performance of DNN models on multiple similar-but-non-identical test datasets. By leveraging the uncertainty estimates generated by these models and matching subsets of datapoints from pairs of test datasets, our approach enables a more nuanced assessment of model performance across datasets with valid differences. Our comprehensive evaluation covered 278 CIFAR-10 and 286 ImageNet models using both original and replicated datasets. Accounting for uncertainty in DNN models, we found that the performance degradation in accuracy was considerably less than previously reported. Specifically, the accuracy gap decreased by 60.83% for CIFAR-10.1, 51.30% for CIFAR-10.2, 57.06% for CINIC-10, and 64.05% for ImageNetV2. These results indicate that DNN model performance remains robust, contrary to earlier findings. Performance degradation likely stems from benign dataset shifts that traditional evaluations fail to detect. Hence, incorporating model-generated uncertainty is crucial for a thorough assessment of DNN behavior.

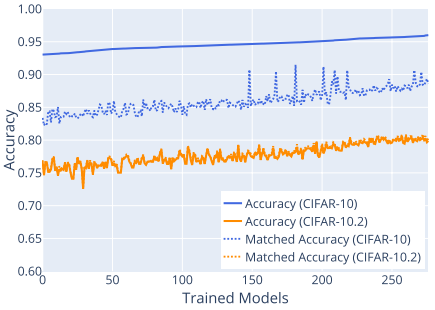
In conclusion, our new evaluation framework represents a step forward in investigating the performance degradation of DNN models on multiple similar-but-non-identical test datasets. While our study utilized softmax probabilities as a baseline for model uncertainty, future research could explore the integration of our framework with more advanced uncertainty estimation techniques to obtain even more accurate and informative assessments of model performance. Furthermore, extending our approach to settings with identifiable covariate shift could provide valuable insights into the robustness and generalization capabilities of DNN models by investigating the consistency of accuracy metrics and uncertainty estimates. The experimental results on CIFAR-10 and ImageNet models highlight the effectiveness of our approach and the importance of incorporating uncertainty information into the evaluation process. Our framework proves particularly useful in complex evaluation scenarios, such as when a DNN model experiences unexpected and substantial performance degradation on test datasets that cannot be fully explained by model


Accuracy
Trained Models
Accuracy (CIFAR-10)
Accuracy (CIFAR-10.1)
Matched Accuracy (CIFAR-10)
Matched Accuracy (CIFAR-10.1)

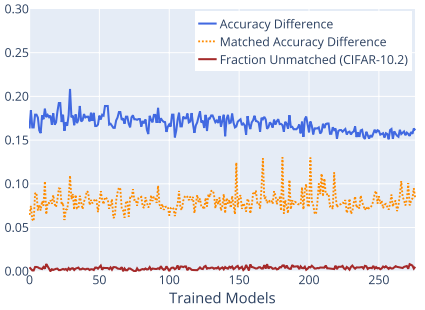
(a)


Trained Models
Accuracy Difference
Matched Accuracy Difference
Fraction Unmatched (CIFAR-10.1)

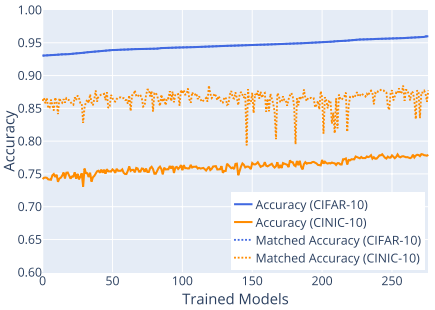
(b)


Accuracy
Trained Models
Accuracy (CIFAR-10)
Accuracy (CIFAR-10.2)
Matched Accuracy (CIFAR-10)
Matched Accuracy (CIFAR-10.2)

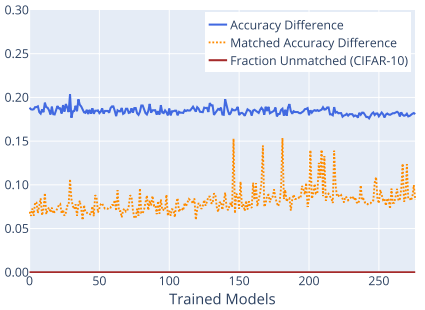
(c)


Trained Models
Accuracy Difference
Matched Accuracy Difference
Fraction Unmatched (CIFAR-10.2)

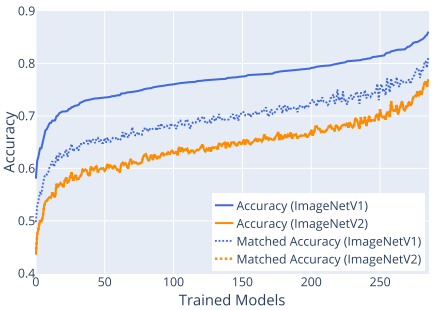
(d)


Accuracy
Trained Models
Accuracy (CIFAR-10)
Accuracy (CINIC-10)
Matched Accuracy (CIFAR-10)
Matched Accuracy (CINIC-10)

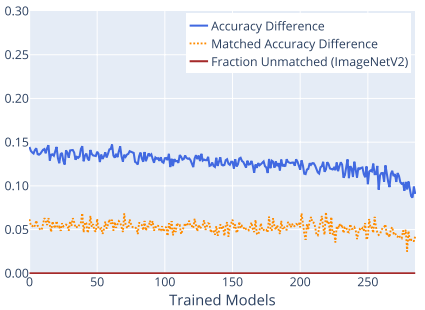
(e)


Trained Models
Accuracy Difference
Matched Accuracy Difference
Fraction Unmatched (CIFAR-10)

(f)


Accuracy
Trained Models
Accuracy (ImageNetV1)
Accuracy (ImageNetV2)
Matched Accuracy (ImageNetV1)
Matched Accuracy (ImageNetV2)

(g)


Trained Models
Accuracy Difference
Matched Accuracy Difference
Fraction Unmatched (ImageNetV2)

(h)

overfitting, and the distinctive characteristics of the datasets cannot easily be identified. In such cases, our uncertainty-aware evaluation method provides valuable insights into the model's behavior and helps to uncover potential sources of performance discrepancies. We believe that our work provides a practical framework for investigating model performance in these challenging situations, and will inspire further research into the development of uncertainty-aware evaluation methods, contributing to the advancement of robust and reliable DNN models for critical real-world applications.

Appendix A Description of dataset, models and additional results

Appendix B Additional characteristics of the matched and unmatched subsets

Factors that can affect the composition of datapoints in the various subsets across all evaluated models include:

- **Model-specific confidence assignment**: Each model assigns confidence scores differently, even for the same inputs. This leads to variability in which datapoints are classified as matched or unmatched across different models. As a result, there is no fixed set of universally unmatched datapoints; for example, an image predicted as "cat" with 90% confidence by Model A may be matched, while the same image predicted with 70% confidence by Model B may remain unmatched. This variability highlights the model-dependent nature of the matching process.
- **Dataset-specific confidence patterns**: Minor differences between the original and replicated datasets could lead to distinct confidence score patterns for each model. For example, similar "cat" images might be predicted with an average confidence of 85% in Dataset A and 80% in Dataset B by the same model, affecting the matching outcomes.
- **The sampling strategy**: Given the disparity in size between the source and target datasets (e.g., ImageNetV1 with 50,000 samples versus ImageNetV2 with 10,000 samples), and the use of sampling without replacement, the order of matching can lead to variability, particularly in the matched source subsets. This procedural aspect alone can account for differences in matched subsets across models or even across multiple runs of the same model.
- **Threshold effects**: The specific threshold (epsilon) used for matching confidence scores can influence the outcomes. Our analysis across different epsilon values (0.01, 0.05, and 0.005) yielded consistent conclusions, thus minimizing the impact of threshold selection on our findings.

References

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review, 78*, 1.https://doi.org/10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2

Bröcker, J., & Smith, L. A. (2007). Increasing the reliability of reliability diagrams. *Weather and Forecasting, 22*(3), 651–661. https://doi.org/10.1175/WAF993.1

Brookhart, M. A., Schneeweiss, S., Rothman, K. J., Glynn, R. J., Avorn, J., & Stürmer, T. (2006). Variable selection for propensity score models. *American Journal of Epidemiology, 163*(12), 1149–1156. https://doi.org/10.1093/aje/kwj149

Cortes, C., DeSalvo, G., & Mohri, M. (2016) Learning with Rejection. In: Proceedings of The 27th International Conference on Algorithmic Learning Theory. Lecture Notes in Computer Science, vol. 9925, pp. 67–82

Chu, J. T. (1965). Optimal decision functions for computer character recognition. *Journal of the ACM, 12*(2), 213–226.

Chen, Y., Li, J., Xiao, H., Jin, X., Yan, S., & Feng, J. (2017) Dual Path Networks. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 4467–4475

Chiappori, P.-A., & Salanié, B. (2016). The Econometrics of Matching Models. *Journal of Economic Literature, 54*(3), 832–861. Publisher: American Economic Association. Accessed 2024-05-26

Caron, M., Touvron, H., Misra, I., Jegou, H., Mairal, J., Bojanowski, P., & Joulin, A. (2021) Emerging Properties in Self-Supervised Vision Transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9630–9640. https://doi.org/10.1109/ICCV48922.2021.00951

Chen, Z., Xie, L., Niu, J., Liu, X., Wei, L., & Tian, Q. (2021) Visformer: The Vision-Friendly Transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 589–598

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021) An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In: Ninth International Conference on Learning Representations

Darlow, L.N., Crowley, E.J., Antoniou, A., & Storkey, A.J. (2018) CINIC-10 is not ImageNet or CIFAR-10. arXiv arXiv:1810.03505

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009) Imagenet: A Large-scale Hierarchical Image Database. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255

Ding, Y., Liu, J., Xiong, J., & Shi, Y. (2019) Evaluation of Neural Network Uncertainty Estimation with Application to Resource-Constrained Platforms. CoRR . arXiv: 1903.02050

Dong, L., Piao, S., & Wei, F. (2022) BEiT: BERT Pre-Training of Image Transformers. In: The Tenth International Conference on Learning Representations

Engstrom, L., Ilyas, A., Santurkar, S., Tsipras, D., Steinhardt, J., & Madry, A. (2020) Identifying Statistical Bias in Dataset Replication. In: Proceedings of the 37th International Conference on Machine Learning, vol. 119, pp. 2922–2932

Gastaldi, X. (2017) Shake-Shake regularization. CoRR arXiv: 1705.07485

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K.Q. (2017) On Calibration of Modern Neural Networks. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 1321–1330

Hein, M., Andriushchenko, M., & Bitterwolf, J. (2019) Why ReLU Networks Yield High-Confidence Predictions Far Away From the Training Data and How to Mitigate the Problem. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 41–50 . https://doi.org/10.1109/CVPR.2019.00013

Hendrycks, D., & Gimpel, K. (2017) A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In: Proceedings of the 5th International Conference on Learning Representations

Ho, D. E., Imai, K., King, G., & Stuart, E. A. (2007). Matching as nonparametric preprocessing for reducing model dependence in parametric causal inference. *Political Analysis, 15*(3), 199–236. https://doi.org/10.1093/pan/mpl013

Han, D., Kim, J., & Kim, J. (2017) Deep Pyramidal Residual Networks. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 6307–6315. https://doi.org/10.1109/CVPR.2017.668

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K.Q. (2017) Densely Connected Convolutional Networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2261–2269. https://doi.org/10.1109/CVPR.2017.243

Hu, J., Shen, L., & Sun, G. (2018) Squeeze-and-Excitation Networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018). https://doi.org/10.1109/CVPR.2018.00745

Hinton, G., Vinyals, O., & Dean, J. (2015) Distilling the Knowledge in a Neural Network. http://arxiv.org/abs/1503.02531

He, K., Zhang, X., Ren, S., & Sun, J. (2016) Deep Residual Learning for Image Recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778. https://doi.org/10.1109/CVPR.2016.90

He, K., Zhang, X., Ren, S., & Sun, J. (2016) Identity Mappings in Deep Residual Networks. In: The European Conference on Computer Vision . https://doi.org/10.1007/978-3-319-46493-0_38

Krizhevsky, A. (2009) Learning Multiple Layers of Features from Tiny Images. Technical report, University of Toronto, Toronto

Kumar, A., Sarawagi, S., & Jain, U. (2018) Trainable Calibration Measures for Neural Networks from Kernel Mean Embeddings. In: Proceedings of the 35th International Conference on Machine Learning, vol. 80, pp. 2805–2814

Krishnan, R., & Tickoo, O. (2020) Improving model calibration with accuracy versus uncertainty optimization. In: Advances in Neural Information Processing Systems, vol. 33, pp. 18237–18248

Lee, D.-H. (2013) Pseudo-Label : The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks.. https://www.semanticscholar.org/paper/Pseudo-Label Accessed 2024-04-30

Lin, T.-Y., Goyal, P., Girshick, R.B., He, K., & Dollár, P. (2017) Focal Loss for Dense Object Detection. 2017 IEEE International Conference on Computer Vision (ICCV), 2999–3007

Lee, K., Lee, K., Lee, H., & Shin, J. (2018) A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. In: Advances in Neural Information Processing Systems, vol. 31

Liang, S., Li, Y., & Srikant, R. (2018) Enhancing the reliability of out-of-distribution image detection in neural networks. In: Proceedings of the 6th International Conference on Learning Representations

Lu, S., Nott, B., Olson, A., Todeschini, A., Vahabi, P., Yair, C., & Schmidt, L. (2020) Harder or Different? A Closer Look at Distribution Shift in Dataset Reproduction. In: Uncertainty and Robustness in Deep Learning Workshop (UDL), ICML

Liu, C., Zoph, B., Shlens, J., Hua, W., Li, L.-J., Fei-Fei, L., Yuille, A.L., Huang, J., & Murphy, K. (2017) Progressive Neural Architecture Search. CoRR **abs/1712.00559**

Mukhoti, J., Kirsch, A., Amersfoort, J.v., Torr, P.H.S., & Gal, Y. (2021) Deterministic Neural Networks with Appropriate Inductive Biases Capture Epistemic and Aleatoric Uncertainty. CoRR **abs/2102.11582**

Müller, R., Kornblith, S., & Hinton, G.E. (2019) When does label smoothing help? In: Wallach, H., Larochelle, H. (eds.) Proceedings of the 33rd International Conference on Neural Information Processing Systems, vol. 422, pp. 4694–4703

Mukhoti, J., Kulharia, V., Sanyal, A., Golodetz, S., Torr, P., & Dokania, P. (2020). Calibrating deep neural networks using focal loss. *Advances in Neural Information Processing Systems, 33*, 15288–15299.

Miller, J.P., Taori, R., Raghunathan, A., Sagawa, S., Koh, P.W., Shankar, V., Liang, P., Carmon, Y., & Schmidt, L. (2021) Accuracy on the Line: on the Strong Correlation Between Out-of-Distribution and In-Distribution Generalization. In: Proceedings of the 38th International Conference on Machine Learning, vol. 139, pp. 7721–7735

Northcutt, C.G., Athalye, A., & Mueller, J. (2021) Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks. In: Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track. https://openreview.net/forum?id=XccDXrDNLek

Nixon, J., Dusenberry, M., Zhang, L., Jerfel, G., & Tran, D. (2019) Measuring Calibration in Deep Learning. http://arxiv.org/abs/1904.01685

Naeini, M.P., Gregory, F.C., & Hauskrecht, M. (2015) Obtaining Well Calibrated Probabilities Using Bayesian Binning. Proceedings of the ... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence 2015, 2901–2907

Northcutt, C., Jiang, L., & Chuang, I. (2021). Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research, 70*, 1373–1411. https://doi.org/10.1613/jair.1.12125

Niculescu-Mizil, A., & Caruana, R. (2005) Predicting good Probabilities with Supervised Learning. In: Proceedings of the 22nd International Conference on Machine Learning, vol. 119, pp. 625–632. https://doi.org/10.1145/1102351.1102430

Pearce, T., Brintrup, A., & Zhu, J. (2021) Understanding Softmax Confidence and Uncertainty. CoRR **abs/2106.04972**

Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., & Bengio, Y. (2015) FitNets: Hints for Thin Deep Nets. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations

Recht, B., Roelofs, R., Schmidt, L., & Shankar, V. (2019) Do ImageNet Classifiers Generalize to ImageNet? In: Chaudhuri, K., Salakhutdinov, R. (eds.) Proceedings of the 36th International Conference on Machine Learning, vol. 97, pp. 5389–5400

Rubin, D.B. (2006) Matched Sampling for Causal Effects. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511810725

Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C. A., Cubuk, E. D., Kurakin, A., & Li, C.-L. (2020). FixMatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems, 33*, 596–608.

Rabanser, Stephan, Günnemann, S., & Lipton, Z. (2019). Failing loudly: An empirical study of methods for detecting dataset shift. *Advances in Neural Information Processing Systems, 32*, 45.

Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A.A. (2017) Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, pp. 4278–4284

Sun, R., & Lampert, C. H. (2020). KS(conf): A Light-Weight Test if a Multiclass Classifier Operates Outside of Its Specifications. *International Journal of Computer Vision, 128*(4), 970–995. https://doi.org/10.1007/s11263-019-01232-x. Accessed 2024-04-30.

Stuart, E. A. (2010). Matching methods for causal inference: A review and a look forward. *Statistical Science: A Review Journal of the Institute of Mathematical Statistics, 25*(1), 1–21. https://doi.org/10.1214/09-STS313

Szyc, K., Walkowiak, T., & Maciejewski, H. (2023) Why Out-of-Distribution detection experiments are not reliable - subtle experimental details muddle the OOD detector rankings. In: Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence, pp. 2078–2088 . ISSN: 2640-3498

Simonyan, K., & Zisserman, A. (2015) Very Deep Convolutional Networks for Large-Scale Image Recognition. In: International Conference on Learning Representations

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. (2021) Training Data-efficient Image Transformers & Distillation Through Attention. In: Proceedings of the 38th International Conference on Machine Learning, vol. 139, pp. 10347–10357

Torralba, A., & Efros, A.A. (2011) Unbiased Look at Dataset Bias. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1521–1528

Tan, M., & Le, Q.V. (2019) EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In: Proceedings of the 36th International Conference on Machine Learning, vol. 97, pp. 6105–6114

Takahashi, R., Matsubara, T., & Uehara, K. (2018) RICAP: Random Image Cropping and Patching Data Augmentation for Deep CNNs. In: Proceedings of The 10th Asian Conference on Machine Learning, vol. 95, pp. 786–798

Thomas, L. E., Yang, S., Wojdyla, D., & Schaubel, D. E. (2020). Matching with time-dependent treatments: A review and look forward. *Statistics in Medicine, 39*(17), 2350–2370. https://doi.org/10.1002/sim.8533

Wightman, R. (2019). PyTorch Image Models. *GitHub. Publication Title: GitHub repository*. https://doi.org/10.5281/zenodo.4414861https://github.com/rwightman/pytorch-image-models

Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017) Aggregated Residual Transformations for Deep Neural Networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp. 5987–5995. https://doi.org/10.1109/CVPR.2017.634

Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., & Yoo, Y. (2019) CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features. In: International Conference on Computer Vision (ICCV)

Zhang, H., Cissé, M., Dauphin, Y.N., & Lopez-Paz, D. (2017) mixup: Beyond Empirical Risk Minimization. CoRR. arXiv: 1710.09412

Zhang, G., Ge, Y., Dong, Z., Wang, H., Zheng, Y., & Chen, S. (2021). Deep high-resolution representation learning for cross-resolution person re-identification. *IEEE Transactions on Image Processing, 30*, 8913–8925.

Authors and Affiliations

Esla Timothy Anzaku[1,2] · Haohan Wang[3] · Ajiboye Babalola[4] · Arnout Van Messem[5] · Wesley De Neve[1,2]

✉ Esla Timothy Anzaku
eslatimothy.anzaku@ugent.be

[1] Department of Electronics and Information Systems, Ghent University, Technologiepark-Zwijnaarde 126, 9052 Ghent, Belgium

[2] Center for Biosystems and Biotech Data Science, Ghent University Global Campus, Munhwa-ro 119-5, Incheon 21985, South Korea

[3] School of Information Sciences, University of Illionois Urbana-Champaign, 501 E. Daniel St. MC-493, Champaign, Chicago 61820-6211, USA

[4] Computational and Data Sciences Department, George Mason Korea, Munhwa-ro 119-4, Incheon 21985, South Korea

[5] Department of Mathematics, University of Liège, 4000 Liège, Belgium