CCC: Continuously Changing Corruptions

Ori Press\textsuperscript{1} Steffen Schneider\textsuperscript{1,2} Matthias Kümmerer\textsuperscript{1} Matthias Bethge\textsuperscript{1}

Abstract

Many existing datasets for robustness and adaptation evaluation are limited to static distribution shifts. We propose a well-calibrated dataset for continuously changing image corruptions on ImageNet scale. Our benchmark builds on the established common corruptions of ImageNet-C and extends them by applying two corruptions at the same time with finer-grained severities to allow for smooth transitions between corruptions. The benchmark contains random walks through different corruption types with different controlled difficulties and speeds of domain shift. Our dataset can be used to benchmark test-time and domain adaptation algorithms in challenging settings that are closer to real-world applications than typically used static adaptation benchmarks.

1. Introduction

Deploying computer vision models into the real world requires robustness against a variety of possible distribution shifts and drifts over long time spans, such as weather, daylight or sensory hardware degradation. Current approaches to robustness and adaptation focus either on ad-hoc robustness or adaptation to fixed noise distributions on shorter timescales.

Currently, models that adapt to their inputs at test time (Schneider et al., 2020; Nado et al., 2020; Wang et al., 2020; Rusak et al., 2021; Wang et al., 2022) are state-of-the-art on classification robustness benchmarks, like ImageNet-C (Hendrycks and Dietterich, 2019). These methods change the model’s weights based on an incoming stream of data. It is therefore necessary to benchmark these approaches not only with noisy images, but with noisy images whose noise gradually changes, much like in the real world.

Though prior work has proposed a variety of ways to model test data that continuously changes, (Hoffman et al., 2014; Bobu et al., 2018; Kumar et al., 2020; Sun et al., 2019; Wang et al., 2022), however only (Wang et al., 2022) model changing shifts on ImageNet scale. Their approach to model changing shifts is to simply concatenate the different noises of ImageNet-C, one after another.

Other approaches gather real world data that necessarily has gradual noise changes (Lomonaco and Maltoni, 2017; Shi et al., 2020; Feng et al., 2019; Han et al., 2021; Sun et al., 2020; Yu et al., 2020). Collecting real world data limits the number of images and noise changes that can feasibly appear in a dataset, as well as the frequency in which the noise changes occur.

\begin{figure}[ht]
\centering
\includegraphics[width=0.8\textwidth]{figure1.png}
\caption{CCC is comprised by combining image corruptions and smoothly varying them over the course of adaptation. The figure shows a sample image from a section of a random walk with a baseline accuracy of 25\%. The dashed red lines indicate where the walk was cut for space.}
\end{figure}
In this work, we propose an ImageNet scale benchmark
to evaluate classification models over long timespans on a
diverse set of noises that continuously change.

The desiderata for the design of our dataset are: (1) building
on the well-established ImageNet-C dataset, (2) adaptation
over long-time scale > 10M images, (3) controlling the
difficulty against a ResNet50 model, (4) a much larger
combinatorial space of possible corruptions and (5) gradually
changing domains over time.

The main ingredient to fulfill these desiderata is the applica-
tion of two different ImageNet-C corruptions at the same
time. By increasing the severity of one corruption type
while decreasing the severity of another corruption type, we
can smoothly transition between corruptions and control the
difficulty precisely.

2. Dataset Preparation

We will now outline the generation procedure of the dataset.
CCC consists of two datasets: The calibration dataset is
comprised of 463 million images along with baseline accu-
cracies computed using a ResNet50 model. The evaluation
dataset consists of random walks through different corrup-
tions and leverages the calibration set to control the baseline
accuracy.

**Calibration Dataset** To generate the calibration dataset,
we take a subset of 5,000 images from the ImageNet val-
ification set. We consider all pairs of the 15 ImageNet-C
test set corruptions. For each corruption, we extend the five
ImageNet-C severities to be more fine-grained by includ-
ing fractions and cover the range of \((0.0, 0.25, \ldots, 5)\) by
interpolating the parameters of the original implementation
functions. We ensure that the difficulty of each corruption
(measured by the top-1 error of our baseline model) mono-
tonically increases with increasing severity. For a given pair
of corruptions and given severities for each corruption, for
each of our 5,000 images, we first apply the first corruption
at its severity and then apply the second corruption at its
severity to the result of the first corruption. In total, our
calibration set is comprised of \(21 \times 21\) combinations of
severities, and \(15 \times 15\) noise pairs, applied to the 5,000 val-
dication images, yielding 463 million images overall. For each
5,000 image subset, we provide the pre-computed ResNet50
baseline accuracy.

**Generating Random Walks** Given the calibration dataset,
we can generate two evaluation datasets that we refer to as
CCC-5k and CCC-50k. The evaluation datasets are gener-
Table 1. Comparison between ImageNet-C, CCC, and CCC-50k

<table>
<thead>
<tr>
<th>Noises</th>
<th>ImageNet-C</th>
<th>CCC</th>
<th>CCC-50k</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severities</td>
<td>15</td>
<td>210</td>
<td>21</td>
</tr>
<tr>
<td>Subsets</td>
<td>5</td>
<td>441</td>
<td>270</td>
</tr>
<tr>
<td>Subset Size</td>
<td>50k</td>
<td>691</td>
<td>50k</td>
</tr>
<tr>
<td>Total Size</td>
<td>3.75M</td>
<td>462M</td>
<td>13.82M</td>
</tr>
</tbody>
</table>

Both datasets rely on the generation of random walks through noise combinations and severities that smoothly transition from one ImageNet-C noise type to the next noise type and so on. To generate such a random walk, we randomly select two noises and vary severities such that the baseline accuracy is as close to the target as possible while transitioning from only one noise to only the other noise. We visualize the process in Figure 3: The path is generated by either decreasing the severity of the first noise by 0.25 (going up) or increasing the severity of the second noise by 0.25 (going right). Given this constraint, we compute the optimal path with mean accuracy closest to our target accuracy by dynamic programming. Once the path through a given noise combination is finished, the noise with severity 0 is randomly replaced, and the process of computing a path for this noise combination repeats.

Each step in the random walk is comprised of 5k, 10k, or 20k images depending on the desired transition frequency. For CCC-5k, we merely repeat the 5k image subset from the calibration set once, twice, or four times. For CCC-50k, we take all images in the ImageNet validation set, corrupt them, and then sample a random subset of the respective size.

3. Evaluation

Comprehensively benchmarking a model is done on several random seeds (which define the order of noises), on several frequencies. In Table 1, we report the number of images required for 9 runs, comprised of 3 random seeds, each run on 3 different frequencies. In this case, each run in CCC-50k is comprised of at least 750k “base” images, that can be repeated as long as necessary. The exact number of “base” images varies from seed to seed, because of the requirement that the “base” random walk start and end on the same noise.

For a given image sequence, we compute the running mean over the last 50k examples. For analysis purposes, this accuracy is reported over the equivalence of 200 “epochs” on the ImageNet validation set. As summary statistics, we report the min, mean and maximum top-1 accuracy over the full adaptation run.

4. Conclusion

We proposed a new benchmark for ImageNet-scale classifiers aimed to benchmark model performance during long-timescale deployment. The benchmark is well-controlled and allows to benchmark models on continuously changing common corruptions, making it particularly interesting for benchmarking test-time and domain adaptation algorithms.

Acknowledgements

We thank Evgenia Rusak for helpful discussions and feedback on the manuscript.

We thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting OP and StS; StS acknowledges his membership in the European Laboratory for Learning and Intelligent Systems (ELLIS) PhD program. StS was supported by a Google Research PhD Fellowship (StS). MB is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project No 390727645 and acknowledges support by the German Research Foundation (DFG): SFB 1233, Robust Vision: Inference Principles and Neural Mechanisms, TP 4, Project No: 276693517. OP and MK were supported by the German Federal Ministry of Education and Research (BMBF) through the Tübingen AI Center (FKZ: 01IS18039A). The authors declare no conflicts of interests.

References


