# Automatically Extracting Scientific Metrics with LLMs: A Case Study of ImageNet Papers

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

In this paper, we introduce a large-scale dataset of papers annotated with their reported Top-1 accuracy on the ImageNet test set, and compare existing and new automatic metric extraction methods, along with a detailed qualitative error analysis. Our study highlights common reporting challenges—such as ambiguous dataset references, table-only metrics, and missing Top-1 values—that drive extraction errors. We curate and release a dataset of 200 manually annotated ImageNet classification papers, larger than prior work, and evaluate our pipeline against both existing approaches and ablated baselines.

## 1 Introduction

Scientific performance metrics – such as top-1 accuracy on ImageNet – play a central role in benchmarking model progress, establishing state-of-the-art (SOTA) claims, and guiding research trends [Bornmann et al., 2021, Barry et al., 2022]. However, metrics are often inconsistently reported across research papers: embedded in tables, mentioned only in captions or abstracts, expressed as error rates, or omitted entirely. As a result, extracting similar metrics at scale remains a largely manual and error-prone process.

In this paper, we present a case study in the automated extraction of top-1 accuracy for image classification reported on ImageNet papers. We construct and release a manually-annotated dataset of 200 ImageNet classification papers—substantially larger than SCILEAD [Şahinuç et al., 2024], which annotated 43 papers across various NLP tasks. By releasing our annotated dataset at `https://anonymous.4open.science/r/imagent-leaderboard-sample-25`, we aim to establish a reproducible benchmark for LLM-assisted scientific metric extraction and support broader efforts in automating literature understanding in machine learning research.

## 2 Annotated dataset

In this section, we describe the construction of our dataset, which is publicly accessible at `https://anonymous.4open.science/r/imagent-leaderboard-sample-25`.

### 2.1 Paper collection

Our dataset originates from an automated collection of publication entries through the PaperWithCode platform, focusing on computer vision papers that report results on the ImageNet dataset for the image classification task. We implemented a simple "try/catch" script to repeatedly query PaperWithCode API for PDFs or arXiv links, automatically skipping entries that lacked valid URLs or produced parsing errors. The step yielded an initial pool of candidate papers, all of which claimed to present top-1 accuracy on ImageNet or its widely recognized variants such as Tiny-ImageNet [Le and Yang, 2015].

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

We programmatically retrieved ImageNet image-classification papers using the `paper_dataset_list` endpoint of the PaperWithCode API, which returns results in its default (unspecified) order.[1] We first selected 12 papers to tune our prompts and then curated another 100 papers to build our development set. For the validation set, we retrieved all papers (focused on the Image Classification task and ImageNet) from PaperWithCodes and selected the top 100 entries, sorted by descending publication date, with preference given to those published in journals afterwards.

### 2.2 Label-verification protocol

From the successfully retrieved documents, we curated a corpus of papers on image classification using the ImageNet dataset. We performed a manual examination of the performance metric reported in each individual paper. Specifically, we identified references to "top-1 accuracy," and pinpointed the corresponding numerical values. We manually annotated each paper with explicit labels such as *(Dataset: ImageNet, Metric: Top-1 Accuracy)*.

### 2.3 Dataset statistics and alignment on ImageNet

Most papers in our corpus report Top-1 Accuracy on the ImageNet dataset. However, many of them were evaluated on *variants* or *subsets* of the ImageNet dataset, such as Tiny-ImageNet [Le and Yang, 2015], ImageNet-100 and etc. For instance, it is a commonplace for evaluations for ILSVRC-2012 or ILSVRC-2015, each of which differ slightly in the number of classes or distribution of images.

Table 2 summarizes the distribution of ground-truth top-1 accuracy presence in both the development and validation sets. In both subsets, only about one-quarter of the papers explicitly report a top-1 accuracy value. This skew reflects a common trend in the literature, where performance metrics—particularly for ImageNet—are often reported only on validation sets, or embedded within complex tables or figures, making automated extraction more challenging. In Fig. 3, we examine the reported metrics from each paper—often referencing *only* a validation subset or using a multi-crop evaluation strategy—and align them to a consistent schema. We attempt to unify various reporting practices; however, directly comparing reported performance metrics across the literature remains challenging.

### 2.4 Comparison with SCILEAD [Şahinuç et al., 2024]

While SCILEAD [Şahinuç et al., 2024] introduces a broad leaderboard spanning multiple NLP tasks and metrics, our dataset is uniquely focused on Image Classification on the ImageNet dataset. We curate a collection of 200 papers that explicitly report top-1 accuracy for the ImageNet classification task, with manually verified ground-truth annotations. We observe 26 papers with metric presence in the development set and 27 in the validation set (demonstrated in Table 2). Moreover, the presence or absence of metrics in our dataset is not artificially balanced or stratified. A detailed qualitative analysis of papers with and without reported metrics is provided in Section 3.2.

## 3 Experimental results and discussion

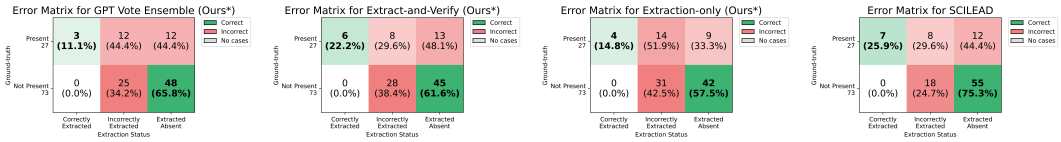
### 3.1 Experimental results

We benchmarked our pipelines—*VOTE-ENSEMBLE*, *EXTRACT-AND-VERIFY*, against the existing SCILEAD baseline in our full annotated validation and development set described in Section 2. Figure 1 presents results on the validation set, and Figure 2 shows results on the development set, reporting presence–correctness confusion matrices for the evaluated extraction systems. Full definitions, additional baselines, and regression metrics are provided in Appendix E.

### 3.2 Qualitative analysis

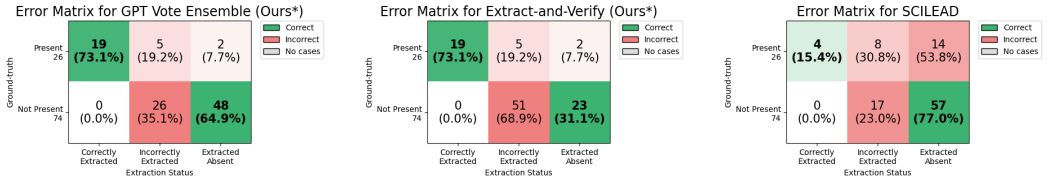
We illustrate our results with a qualitative analysis of easy and difficult cases, highlighting the challenges from the evaluated extraction systems. We extend our qualitative analysis in Appendix F.

[1] The API documentation does not state the sorting criterion; see `https://paperswithcode-client.readthedocs.io/en/latest/api/client.html`.




(a) (b) (c) (d)

Figure 1: Presence–correctness confusion matrices on the validation set in comparison with extraction quality across four methods: (a) *VOTE-ENSEMBLE (Ours*)*, (b) *EXTRACT-AND-VERIFY (Ours*)*, (c) *EXTRACT-only (Ours*)*, and (d) SCILEAD. Each matrix summarizes extraction outcomes conditioned on whether a Top-1 accuracy value is present in the ground-truth (*rows*) and whether the system produced an extraction (*columns*). Green cells indicate correct extractions, red cells indicate incorrect or hallucinated values, and blank cells indicate non-extraction when ground truth is absent.




(a) GPT Vote Ensemble (Ours*) (b) Extract-and-Verify (Ours*) (c) SCILEAD

Figure 2: Error matrices comparing extraction quality across methods on the development set.

### 3.2.1 Straightforward ("easy") examples

- **Abstract, 1807.11626v3** (Fig. 5): "*On the ImageNet classification task, our **MnasNet** achieves 75.2% top-1 accuracy . . .*" *—A single numeric value appears in the abstract; the metric name and dataset are explicit.*

### 3.2.2 Difficult ("challenging") Examples

- **Alternate dataset variant** (Fig. 7): "*V-CapNet reaches 99.83% validation accuracy on the **Natural Images** dataset.*" *—Dataset differs from canonical ImageNet; the numeric value should be excluded from extraction.*
- **Table-only Top-1 value** (Fig. 10): ImageNet top-1 accuracy (76.82%) is given only in a table, with no reference in the main text. *—Requires table reading; plain text search may be error-prone.*
- **Multiple candidate Top-1 values** (Fig. 11): A table in 1909.13863v1 lists four *Case* rows with different top-1 accuracies (55.5–57.1%). *—Require an investigation on which value is the main result or flag ambiguity.*
- **Top-1 accuracy omitted, only Top-5 present** (Fig. 12): Sentence in 1807.10119v3 reports "top-5 accuracy drops slightly . . ." while never stating Top-1. *—Require returning "missing" rather than hallucinate a Top-1 value.*
- **Test vs. Validation ambiguity** (Fig. 17): "*Classification performance comparison on ImageNet (single crop, single model)... "* *—Mentions classification on ImageNet without specifying test/validation split, creating ambiguity on the split.*
- **Top-1 metrics embedded in large metric tables** (Fig. 9, Fig. 22–31, with additional examples in Fig. 32–35, detailed in Section F.1): The top-1 accuracy for ImageNet appears within tables that contain dozens of entries, dense formatting, and mixed dataset contexts. These settings present challenges for automated extraction systems, due to ambiguous column headers, multi-dataset rows, or inconsistent labeling. We highlight ten such cases:
  - **Ambiguous metric header** (Fig. 22): A table contains values such as 85.6 and 89.5 without specifying the evaluation split (validation/test) or whether the numbers correspond to top-1 accuracy. Systems returned mismatched values that failed to align with the ground truth (87.2).
  - **Cross-dataset overload** (Fig. 23): The table mixes results for ImageNet-1K, CIFAR-100, and TinyImageNet. The extracted value (82.0) does not correspond to the correct ImageNet Top-1 metric (85.9), highlighting difficulties in aligning rows with the intended benchmark.

*—Extractors must jointly reason over model–metric–dataset alignment, disambiguate unlabeled columns, and avoid numeric heuristics. Without semantic parsing or table structure awareness, these large tables often lead to hallucinations or near-miss errors.*

**Key take-aways** Easy cases share three traits: a single accuracy value, explicit dataset naming, and standard wording. Failures occur when authors (i) report on validation instead of test set, (ii) report on ImageNet variants or sampled subsets, (iii) place numbers only in supplementary material, or (iv) use alternate metrics such as error rates or Top-5 accuracy. These observations guided our rule-based post-processing (for error-to-accuracy conversion) and the heuristics that flag ambiguous dataset references.

#### 3.2.3 Cross-system top-1 accuracy extraction comparison on selected failure cases

We analyze additional representative failure cases across systems, with extended examples provided below.

**Top-1 metrics reported on validation set only** (Fig. 21): A table presents classification and localization error rates (Top-1, Top-5) on ILSVRC-15 *validation set*. *—Top-1 accuracy must be inferred by subtracting from 100%; Top-1 scores on test set is never given.*

**ImageNet was used in pretraining in the paper**

- **Non-ImageNet dataset with pretrained model** (Fig. 20): "*Caltech-256: 84.7 (ImageNet-CLS), 76.7 (OpenImages)*" *—Although the model is pretrained on ImageNet, evaluation is done on Caltech-256; such results should not be extracted as ImageNet scores.*

**Top-1 metrics embedded in large metric tables** (Fig. 26, Fig. 27 and Fig. 28) Top-1 accuracy for ImageNet frequently appears within large, multi-column tables featuring dense formatting and mixed dataset benchmarks. The complex layouts pose challenges for automated extraction systems, particularly when headers are ambiguous or dataset-metric alignment is unclear. Below, we present three additional annotated examples.

- **Missing label context** (Fig. 26): Extractors failed to identify that 80.99 was the correct ImageNet Top-1 accuracy. Instead, they returned 77.85 due to ambiguity in model-type alignment and lack of direct sentence reference.
- **Column overload in architecture benchmarking** (Fig. 27): Dense layout with models, FLOPs, and multiple accuracy metrics makes correct alignment difficult. Ground truth (79.96) was not captured by any system.
- **Grouped ViT-family entries with sparse labels** (Fig. 28): The ground-truth Top-1 of 79.6 is lost among model configurations. Systems instead hallucinated higher values (e.g., 82.0, 84.2) drawn from unrelated rows.

| Paper | Ground-truth | VOTE-ENSEMBLE | EXTRACT-AND-VERIFY | SCILEAD | EXTRACT-only |
|---|---|---|---|---|---|
| `1610.02391v4.pdf` (Fig. 21) | 69.62 | – | 70.58 | 70.58 | – |
| `2505.14124v1.pdf` (Fig. 26) | 80.99 | – | 77.85 x | – | 77.85 x |
| `2412.15077v1.pdf` (Fig. 27) | 79.96 | – | – | – | 68.28 x |
| `2411.15241v1.pdf` (Fig. 28) | 79.6 | 84.22 x | – | 81.9 x | 82.0 x |

Table 1: Cross-system Top-1 Accuracy extraction comparison on selected failure cases. Each system either outputs an incorrect value (denoted by x) or abstains ("–"). These examples are highlighted in the qualitative analysis (Sec. 3.2).

## 4 Conclusion

We presented a case study of extracting scientific metrics from the scientific literature using a historically significant metric: top-1 accuracy reported on the ImageNet test set. We release a large annotated dataset (substantially larger than previous annotated datasets of 43 papers) and present detailed qualitative error analysis.

We experimented with prior work as well as several methods based on recent ideas such as self-critique. Expanding the labelled dataset would allow for enough statistical power to compare different methods.

# References


Erin S Barry, Jerusalem Merkebu, and Lara Varpio. Understanding state-of-the-art literature reviews. *Journal of graduate medical education*, 14(6):659–662, 2022.

Lutz Bornmann, Robin Haunschild, and Ruediger Mutz. Growth rates of modern science: A latent piecewise growth curve approach to model publication numbers from established and new literature databases, 2021. URL https://arxiv.org/abs/2012.07675.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

Xinyun Chen, Maxwell Lin, Nathanael Schärli, and Denny Zhou. Teaching large language models to self-debug, 2023. URL https://arxiv.org/abs/2304.05128.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

Yann Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback, 2023. URL https://arxiv.org/abs/2303.17651.

Furkan Şahinuç, Thy Thy Tran, Yulia Grishina, Yufang Hou, Bei Chen, and Iryna Gurevych. Efficient performance tracking: Leveraging large language models for automated construction of scientific leaderboards. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7963–7977, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.453. URL https://aclanthology.org/2024.emnlp-main.453/.

Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. On the self-verification limitations of large language models on reasoning and planning tasks, 2024. URL https://arxiv.org/abs/2402.08115.

Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao. Large language models are better reasoners with self-verification, 2023. URL https://arxiv.org/abs/2212.09561.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, et al. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*, 2022.

| | | | Test | | 10-crop Validation | Single-Model Validation | |
|---|---|---|---|---|---|---|---|
| File Name | Paper Name | Model | Top-1 | Top-5 A | Top 1 | Top-1 | Top-5 A |
| 1512.03385v1.pdf | Deep Residual Learning for Image R | ResNet-18 | | | | | |
| 1512.03385v1.pdf | Deep Residual Learning for Image R | ResNet-34 | | | | | |
| 1512.03385v1.pdf | Deep Residual Learning for Image R | ResNet-50 | | | | | |
| 1512.03385v1.pdf | Deep Residual Learning for Image R | ResNet-101 | | | | | |
| 1512.03385v1.pdf | Deep Residual Learning for Image R | ResNet-152 | | 96.43% | 78.57% | 80.57% | |
| 1703.09844v5.pdf | Multiscale Dense Networks for Resc | MSDNet | 75% | - | | | - |
| 1803.00942v3.pdf | Not All Samples Are Created Equal: | ResNet-50 | - | - | | - | - |
| 1807.10108v5.pdf | Effects of Degradations on Deep Ne | V-CapsNet | | - | | 99.83% | - |
| 1807.10119v3.pdf | A Unified Approximation Framework | AlexNet | - | 80% | | - | |
| 1807.11164v1.pdf | ShuffleNet V2: Practical Guidelines | ShuffleNet v2-50 | 77.20% | - | | | - |
| 1807.11254v2.pdf | Extreme Network Compression via F | ResNet34 | 64.75 | 64.3 | | | |
| 1807.11459v1.pdf | Improving Transferability of Deep Ne | ResNet-27 | - | - | | - | - |
| 1807.11626v3.pdf | MnasNet: Platform-Aware Neural Ar | MnasNet | 75.20% | - | | | - |
| 1909.11155v1.pdf | Anchor Loss: Modulating Loss Scale | ResNet-50 | 76.82% | 93.03% | | | |
| 1909.13863v1.pdf | XNOR-Net++: Improved Binary Neura | Binary ResNet-18 | 57.10% | 79.90% | | | |
| 1909.13863v1.pdf | XNOR-Net++: Improved Binary Neura | Binary AlexNet | 46.90% | 71.00% | | | |
| omniVec_2023.pdf | OmniVec: Learning robust represent | OminiVec (FT) | 92.40% | - | | | - |

Figure 3: ground-truth table for a selected set of 12+ papers, recording Top-1/Top-5 values from the test set, validation set, or a specific multi-crop procedure (e.g., 10-crop validation from Krizhevsky et al., 2017). "–" and space denotes that no explicit metric was identified for that field.

# A Related work

Our work builds on and integrates a sequence of ideas, including diverse prompting techniques and the self-critique paradigm. Prior work such as SCILEAD [Şahinuç et al., 2024] has advanced the construction of scientific leaderboards through an automated extraction pipeline. Our contribution lies in applying these techniques—beginning with structured prompts incorporating verification and ensemble ideas—for an automated extraction of scientific metrics from research papers.

In few-shot prompting, also referred to as "in-context learning," [Brown et al., 2020], an LLM is provided with a prompt consisting of multiple examples of the target task, each in the form of input-output pairs.

Query self-refinement is argued to enhance the initial outputs of large language models (LLMs) [Madaan et al., 2023]. Self-refinement is inspired by the way humans revise their written text. This method consists of an iteratively feedback-driven process that improves the initial responses generated by LLMs.

The Least-to-Most prompting framework was introduced by Zhou et al. [2022], which decomposes reasoning tasks into structured subproblems to improve performance. Each subproblem is solved in sequence with subsequent steps conditioned on previous answers. The least-to-most prompting enables LLMs to generalize for problems with more difficulties than those present in the prompts.

SCILEAD introduced an LLM-based method for the automatic constructions of scientific leaderboards [Şahinuç et al., 2024]. SCILEAD contributes a manually curated dataset of leaderboards drawn from 43 scientific papers and proposes an extraction schema based on task–dataset–metric (TDM) triples, where each triple represents an extraction task, the associated dataset, and the reported evaluation metric. The work exhaustively annotates individual papers by labeling all unique TDM combinations along with their respective top-reported results.

Several recent studies have explored the concept of direct self-critique by LLMs [Stechly et al., 2024, Weng et al., 2023, Chen et al., 2023]. Inspired by human cognitive processes, these approaches leverage the intuition that verifying or critiquing an answer is typically easier or fundamentally different from generating it from scratch. Hence, the strategies demonstrate potential to improve the overall quality of outputs [Stechly et al., 2024]. In a standard self-critique pipeline, an LLM first generates an answer and subsequently receives its own response as input, along with explicit prompting instructions to critique, refine, or revise the original answer. The self-critique loop iterates and then stops at a predefined stopping criterion.

Our work builds upon these ideas by incorporating explicit verification steps, enabling cross-checking and refinement of extracted performance metrics.

| Metric Presence | Development Set | Validation Set |
|---|---|---|
| Ground-truth Present | 26 | 27 |
| Ground-truth Absent | 74 | 73 |
| Total | 100 | 100 |

Table 2: Distribution of ground-truth Top-1 Accuracy presence across the 100-paper development and validation sets. Many papers do not explicitly report Top-1 Accuracy, often deferring such results to supplemental materials or validation splits.

# B Extraction experiments

We describe our extraction experiments for *EXTRACT-AND-VERIFY* in detail. We first introduce the *VOTE-ENSEMBLE* style of prompting for aggregating predictions across paper sections, and then describe how *EXTRACT-AND-VERIFY* builds upon it with an additional verification step.

Let $n$ denote the number of text segments derived from each paper (e.g., abstract, results, conclusion), and $k$ the number of prompt attempts per section for diverse extraction outputs. The value of $n$ is computed dynamically for each paper using a lightweight chunking function, which groups every three consecutive pages into a single segment. We set $k = 5$ in all experiments.

***VOTE-ENSEMBLE* prompting** We begin by dividing each paper into $n$ segments—typically corresponding to sections such as the abstract, experimental results, and conclusion. For each segment, we apply an extraction prompt that instructs the LLM to extract a top-1 accuracy reported on ImageNet and its corresponding source sentence from literature. This extraction process is repeated $k$ times per segment to capture various responses. The resulting $k$ sentence–accuracy pairs for i-th segment are then aggregated using our *VOTE-ENSEMBLE* strategy, which selects the pair most frequently occurring as the final output. The prompt used in this step is shown in Section C.1.

***EXTRACT-AND-VERIFY* prompting** We introduced a verification phase during the extraction process of our *EXTRACT-AND-VERIFY*. For each section, the LLM re-evaluates each extracted sentence–accuracy pair in the context of the original PDF page to verify whether the sentence appears in the text and whether the extracted value explicitly corresponds to a top-1 accuracy on ImageNet. The verification is applied across the $k$ extractions generated during the ensemble step. The prompt template used for verification is demonstrated in Section C.2.

To isolate the effect of ensembling and verification, we introduce *EXTRACT-only*—a minimal baseline where extraction is performed with a single prompt iteration ($k = 1$) based on VOTE-ENSEMBLE. It enables the independent assessment of each system component.

# C Prompt examples

## C.1 *VOTE-ENSEMBLE prompt demonstration*

Prompt Input

```
Find the accuracy value associated with most common sentences from the list of
sentences and accuracies. Only output the accuracy value.

Example 1:
'Sentence: "ImageNet1K Top-1 Accuracy ViT 88.5 89.1 88.6 92.4": 92.4', 'Sentence:
"SSv2 Top-1 Accuracy ViViT 65.4 68.6 80.1 85.4 ImageNet1K Top-1 Accuracy ViT 88.5 89.1
88.6 92.4 Sun RGBD Top-1 Accuracy Simple3D-former 57.3 62.4 71.4 74.6"Accuracy: 92.4',
'Sentence: "ImageNet1K Top-1 Accuracy ViT 88.5 89.1 88.6 92.4": 92.4', 'Sentence:
"SSv2 Top-1 Accuracy ViViT 65.4 68.6 80.1 85.4 ImageNet1K Top-1 Accuracy ViT 88.5
89.1 88.6 92.4 Sun RGBD Top-1 Accuracy Simple3D-former 57.3 62.4 71.4 74.6": 74.6',
Expected Output: 92.4

Example 2:
'Sentence: "SSv2 Top-1 Accuracy ViViT 65.4 68.6 80.1 85.4 ImageNet1K Top-1 Accuracy
ViT 88.5 89.1 88.6 92.4 Sun RGBD Top-1 Accuracy Simple3D-former 57.3 62.4 71.4
74.6": 74.6', 'Sentence: "ImageNet1K Top-1 Accuracy ViT 88.5 89.1 88.6 82.4": 82.4',
'Sentence: "SSv2 Top-1 Accuracy ViViT 65.4 68.6 80.1 85.4 ImageNet1K Top-1 Accuracy
```

ViT 88.5 89.1 88.6 82.4 Sun RGBD Top-1 Accuracy Simple3D-former 57.3 62.4 71.4 74.6": 82.4', Expected Output: 82.4

**Example 3:**
'Sentence: "It's not mentioned top-1 accuracy on ImageNet": NA', 'Sentence: "-": NA', 'Sentence: "It's not mentioned top-1 accuracy on ImageNet": NA', 'Sentence: "-": NA', Expected Output: NA

**Example 4:**
'Sentence: "It's not mentioned.": NA', 'Sentence: "-": NA', 'Sentence: "It's mentioned top-5 accuracy on ImageNet": NA, 'Sentence: "Cocoa 23.3 21.2": 23.3', Expected Output: NA

Now extract the accuracy value associated with most common sentences, sentences$_and_accuracies$

Expected Output:

### C.2 *EXTRACT-AND-VERIFY* prompt demonstration

Prompt Input

**INPUTS**
• page: full text of one PDF page, page
• pair: "Sentence:<sentence> : <accuracy>", a_sentence_and_accuracy

**TASK**
Check whether the sentence appears in the page (with minor formatting variations), and whether the accuracy is a valid Top-1 ImageNet value.

- If both sentence and accuracy are correct and found in the page, return the pair unchanged. - If the accuracy is incorrect but a correct one exists, return the corrected pair. - If no Top-1 ImageNet accuracy is found, return an empty string.

**IMPORTANT**
• Only match Top-1 accuracy on ImageNet (not Top-5, CIFAR, COCO, etc.)
• Output exactly one line, or nothing at all.

**OUTPUT**
Sentence:<sentence> : <accuracy>

## D Error analysis metrics summarization

### D.1 Regression metrics definitions

We present the regression metrics definitions in Table 3.

### D.2 Presence–correctness matrix analysis

#### D.2.1 12-paper set

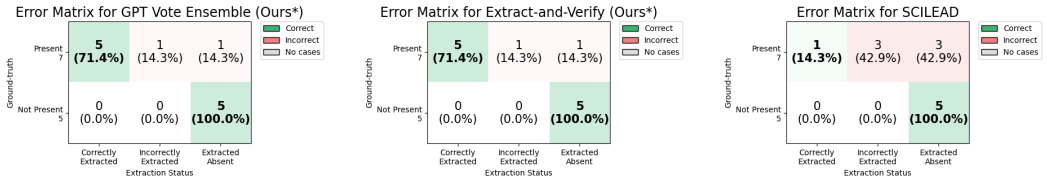
Figure 4 presents the error analysis in the form of confusion matrices along two axes:

1. whether the paper reports a Top-1 Accuracy value on ImageNet (*row*)
2. whether the system extracts such a value (*column*)

The counts are computed over the 12 manually inspected papers listed in Table 7.

Table 3: Evaluation metrics and error categorization for extraction tasks, with matrix axis definitions and term abbreviations.

| Category | Metric Description |
|---|---|
| **Presence–Correctness Matrix Axes** | |
| **Row: Ground-truth Presence** | Whether the paper reports a Top-1 Accuracy value on ImageNet. Two possible states:<br>**Ground-truth Present** – the metric is reported in the pape.<br>**Ground-truth Absent** – the metric is not reported. |
| **Column: System Extraction** | Whether the system extracts a Top-1 Accuracy value. Three possible outputs:<br>**Correctly Extracted/Extract Match(EM)** – value is correctly extracted and matches the ground-truth.<br>**Incorrectly Extracted** – value is extracted but incorrect (e.g., hallucinated or wrong metric).<br>**Extracted Absent** – system does not extract any value. |
| **Classification-style Metrics** | |
| **#Extract Match (EM)** | Exact Match (EM) is defined as a correct extraction in which the extracted top-1 accuracy value exactly equals the ground-truth value recorded in the paper. EMs are only counted when the ground-truth value is present, and the extracted value aligns numerically with it (e.g., 76.82 matches 76.82 exactly). |
| **#Incorrect Extractions** | the number of incorrect extractions when ground-truth is present. |
| **Regression-style Metrics** | |
| | Let $\hat{y}_i$ denotes extracted values, $y_i$ denotes ground-truth values and $n$ denotes the total number of papers where both ground-truth and extractions are present: |
| **MAE** | **Mean Absolute Error** between extracted and ground-truth numeric values: $\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\lvert\hat{y}_i - y_i\rvert$. |
| **RMSE** | **Root Mean Square Error**: $\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}$. |




(a) GPT Vote Ensemble (Ours*) (b) Extract-and-Verify (Ours*) (c) SCILEAD

Figure 4: Error matrices comparing extraction quality across methods.

# E Full experiment results

## E.1 Runtime discussion & API rate limits

We report the runtime and cost to process one PDF on a single 8-core CPU with a 100 Mbps network link to the OpenAI endpoint. The user tier 1 imposes `90 000` tokens min$^{-1}$ and `3 500` requests min$^{-1}$. Our pipeline uses a single `gpt-4o` call per sections (consisting of a few pages), averaging ~1 650 input tokens and 120 output tokens. We apply a dynamic safety delay time based on the characteristics of each paper between calls of OpenAI requests. As a result, the average processing time per document is approximately 15 minutes—making the pipeline more computationally intensive than initially expected. We evaluate the performance of our proposed extraction pipelines, *VOTE-ENSEMBLE*, *EXTRACT-AND-VERIFY*, against the existing SCILEAD baseline in our full annotated validation and development set described in Section 2. We then report on regression-based metrics—quantifying how close the extracted values are numerically. The details of metric definitions can be found in Section D.

| System | #Extract Match ↑ | #Incorrect Extractions ↓ | MAE ↓ | RMSE ↓ |
|---|---|---|---|---|
| *VOTE-ENSEMBLE (Ours*)* | 3 | 12 | 19.633 | 35.672 |
| *EXTRACT-AND-VERIFY (Ours*)* | 6 | 8 | 3.033 | 4.500 |
| *EXTRACT-only (Ours*)* | 4 | 14 | **2.831** | **4.103** |
| *SCILEAD* | **7** | 8 | 5.026 | 11.027 |

Table 4: Regression metrics comparing *VOTE-ENSEMBLE*, *EXTRACT-AND-VERIFY*, *EXTRACT-only*, and SCILEAD on the 100-paper validation set. *EXTRACT-only* represents a minimal baseline where extraction is performed using a single section without voting.

### E.2 100 Validation set with *EXTRACT-only* baseline

We compare our proposed methods, *VOTE-ENSEMBLE*, *EXTRACT-AND-VERIFY*, *EXTRACT-only* against SCILEAD on our 100-paper validation set. Figure 1 shows the presence–correctness confusion matrices for four extraction systems evaluated in the validation set.

Among the four systems, SCILEAD achieved the highest number of exact matches (7), followed by *EXTRACT-AND-VERIFY* (6), *EXTRACT-only* (4), and *VOTE-ENSEMBLE* (3). Notably, *EXTRACT-AND-VERIFY* exhibited lower regression errors compared to SCILEAD and *VOTE-ENSEMBLE*, despite recovering slightly fewer exact matches. As reported in Table 4, both *EXTRACT-AND-VERIFY* and *EXTRACT-only* achieved the lowest Mean Absolute Error (MAE) of 3.211 and RMSE of 4.550. In contrast, *VOTE-ENSEMBLE* reported high numeric errors among its 12 incorrect extractions, yielding an MAE of 19.633 and RMSE of 35.672. SCILEAD showed moderate regression performance (MAE: 5.026, RMSE: 11.027) but had a higher count of incorrect extractions (8). While these results suggest performance differences across systems, we caution that the observed variations may not be statistically significant due to the limited size of the validation set. To better understand the impact of prompt aggregation and verification, we include an additional *EXTRACT-only* baseline with details in Section E.4.

### E.3 Development set results

We assess our proposed methods, *VOTE-ENSEMBLE*, *EXTRACT-AND-VERIFY* against SCILEAD on our 100-paper development set. Figure 2 presents presence–correctness confusion matrices comparing *VOTE-ENSEMBLE*, *EXTRACT-AND-VERIFY*, and SCILEAD on our development set. To quantify performance, we compute regression metrics over incorrect extractions—cases where both the ground-truth and predictions are present, but the extracted values are incorrect. Table 5 reports the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) for these incorrect extractions.

Both of our methods, *VOTE-ENSEMBLE* and *EXTRACT-AND-VERIFY*, correctly extracted 19 top-1 accuracy values on ImageNet (73.1%), outperforming SCILEAD, which achieved 4 exact matches [2] (15.4%). While both of our methods reported 5 incorrect extractions, *EXTRACT-AND-VERIFY* showed higher regression error in terms of regression errors among them. As shown in Table 5, *EXTRACT-AND-VERIFY* achieved a higher Mean Absolute Error (MAE) of 1.632 and RMSE of 6.029, compared to 0.642 MAE and 2.112 RMSE for *VOTE-ENSEMBLE*. SCILEAD presented higher error (MAE: 14.482, RMSE: 28.800). The results are based on a limited development set and may not be statistically significant.

### E.4 *EXTRACT-only* baseline result

To better understand the impact of prompt aggregation and verification, we include an additional *EXTRACT-only* baseline. The method corresponds to setting $k = 1$ in *VOTE-ENSEMBLE*. While *EXTRACT-only* underperformed SCILEAD in terms of exact match (4 vs. 7), it achieved lower regression error, with an MAE of 2.831 and RMSE of 4.103. Again, we reiterate that our findings are based on a small sample and suggest they be interpreted with caution.

## F Extensions to Qualitative Analysis

### F.1 More Cross-System Top-1 Extraction Comparison on Selected Failure Cases

We analyze additional representative failure cases across systems, with extended examples provided below.

[2] Exact Match (EM) refers to a correct extraction where the extracted value exactly matches the ground-truth value present in the paper.

| System | #Extract Match ↑ | #Incorrect Extractions ↓ | MAE ↓ | RMSE ↓ |
|---|---|---|---|---|
| *VOTE-ENSEMBLE (Ours*)* | 19 | 5 | 0.642 | 2.112 |
| *EXTRACT-AND-VERIFY (Ours*)* | 19 | 5 | 1.632 | 6.029 |
| SCILEAD | 4 | 8 | 14.482 | 28.800 |

Table 5: Regression metrics comparing *VOTE-ENSEMBLE*, *EXTRACT-AND-VERIFY* and SCILEAD on development set.

**Top-1 metrics reported on validation set only** ( Fig. 15, Fig. 16, Fig. 21):

- **Top-1 scores on validation set only** (Fig. 15): "*We acquire better classification results on complex validation set . . . "* *—Top-1 scores on validation reported; Top-1 scores on test set is never given.*
- **Top-1 scores on validation set only** (Fig. 16): "*. . . top-1 and top-5 error rates on the ImageNet* ***validation set*** *. . . "* with Top-1 = 22.15% for DenseNet-264. *—Top-1 must be derived from the error rate (100% - 22.15%), and validation set must be interpreted correctly.*
- **Top-1 scores on validation set only** (Fig. 21): A table presents classification and localization error rates (Top-1, Top-5) on ILSVRC-15 *validation set*. *—Top-1 must be inferred by subtracting from 100%; Top-1 scores on test set is never given.*

**ImageNet was pretrained in the paper** (Fig. 18, Fig. 19, Fig. 20): *—Requires filtering out such examples as false positives despite mentioning ImageNet.*

- **Non-ImageNet dataset with pretrained model** (Fig. 18): "*We obtained an accuracy of 91.66% and 78.01% for the CALTECH 101 and neuromorphic CALTECH 101 datasets respectively.*" *—Reports accuracy while using ImageNet-pretrained networks, but on non-ImageNet datasets (CALTECH 101); must be excluded.*
- **Table-only accuracy, but not ImageNet** (Fig. 19): Table lists results like "85.3" under DA–ADAGE Incremental for MNIST-M and SVHN. *—Metrics shown in table format but pertain to other domains (MNIST-M, SVHN); dataset mismatch with ImageNet.*
- **ImageNet-pretrained model on different datasets** (Fig. 20): "*Caltech-256: 84.7 (ImageNet-CLS), 76.7 (OpenImages)*" *—Although the model is pretrained on ImageNet, evaluation is done on Caltech-256; such results should not be extracted as ImageNet scores.*

**Top-1 metrics embedded in large metric tables (continued)** (Fig. 9, Fig. 22–31, and Fig. 32–35): Top-1 accuracy for ImageNet frequently appears within large, multi-column tables featuring dense formatting and mixed dataset benchmarks. The complex layouts pose challenges for automated extraction systems, particularly when headers are ambiguous or dataset-metric alignment is unclear. Below, we present four additional annotated examples.

- **Misaligned extraction due to unlabelled columns** (Fig. 25): Although the table includes Top-1 classification accuracy (81.02), extractors reported nearby but incorrect values (e.g., 76.1 or 79.5), likely due to metric misinterpretation.
- **Under-specified table with missing axis labels** (Fig. 30): The ImageNet Top-1 accuracy (76.2) is never directly labeled. Extractors returned 72.2, reflecting structural ambiguity in the source format.
- **Multiple candidate metrics in same table** (Fig. 31): The Top-1 accuracy (78.1) is buried among other results. Extractors selected 70.0, suggesting overreliance on numerical proximity rather than structured alignment.
- (Fig. 32): "*Comparison of Top-1 accuracy across various methods on the ImageNet dataset...*"

  Top-1 scores are shown across architectures like RN50 and ViT-B/16.
- (Fig. 33): "*ImageNet: 74.1*" appears in a large benchmark spanning multiple datasets.

  Despite being explicitly labeled, the metric is embedded among heterogeneous datasets, making it unclear. Systems may incorrectly associate the wrong metric with the wrong dataset.
- (Fig. 34): "*Image classification results (Acc, %) on ImageNet.*"

  The table mixes logit-based and feature-based methods across two settings. Extractors must infer correct Top-1 values from rows and columns with inconsistent grouping and abbreviations. Without semantic understanding of headers, incorrect matches frequently occur.
- (Fig. 35): "*ImageNet: 62.78*" appears in a row alongside CUB200, EuroSAT, and other datasets.

| Paper | Ground-truth | VOTE-ENSEMBLE | EXTRACT-AND-VERIFY | SCILEAD | EXTRACT-only |
|---|---|---|---|---|---|
| 2501.10640v2.pdf (Fig. 22) | 87.2 | 85.6 x | – | 83.9 x | 89.5 x |
| 2501.07783v1.pdf (Fig. 23) | 85.9 | 82.0 x | 82.1 x | – | 82.0 x |
| 2505.14062v1.pdf (Fig. 24) | 83.0 | – | – | 67.5 x | – |
| 2410.08407v2.pdf (Fig. 25) | 81.02 | – | 76.1 x | 79.51 x | 76.1 x |
| 2412.02366v3.pdf (Fig. 29) | 78.73 | 73.3 x | – | 65.8 x | 77.23 x |
| 2410.10773v1.pdf (Fig. 31) | 78.1 | – | 70.0 x | – | – |
| 2504.08710v1.pdf (Fig. 30) | 76.2 | – | 72.2 x | – | – |
| 2412.20110v3.pdf (Fig. 32) | 74.23 | 82.0 x | 66.17 x | 36.88 x | 66.17 x |
| 2503.12206v2.pdf (Fig. 33) | 74.1 | – | 83.44 x | – | – |
| 2412.08139v1.pdf (Fig. 34) | 73.69 | 82.2 x | – | 72.49 x | 71.35 x |
| 2412.11917v3.pdf (Fig. 35) | 62.78 | 71.89 x | – | – | 63.31 x |

Table 6: Cross-system Top-1 Accuracy extraction comparison on selected failure cases. Each system either outputs an incorrect value (denoted by x) or abstains ("–"). These examples are highlighted in the qualitative analysis (Sec. 3.2 and Sec. F.1).

# G Limitations

While introducing a larger annotated dataset than prior work, several limitations remain in our study. First, despite being larger than previous work, our dataset may still lack sufficient statistical power to demonstrate significant performance differences across various extraction systems. Moreover, our use of the PapersWithCode API for sampling—given its default ordering—may introduce sampling bias.

While ImageNet has historically served as a foundational benchmark dataset, it may not fully capture the diversity of datasets or evaluation metrics in computer vision or the broader computer science community. Therefore, our findings may not generalize to other tasks, domains, or scientific fields with different reporting conventions and metric structures.

# H Ethics statement

Our work introduces a large-scale dataset of papers annotated with their reported top-1 accuracy on the ImageNet test set. All PDF papers in our study are publicly available on arXiv, which permits fair use and supports responsible, reproducible, and transparent scientific research practices. All annotation work was performed by the authors.

Our released dataset contains entries consisting of arXiv identifiers and their corresponding labeled Top-1 accuracy values on ImageNet. The intended use of our released dataset is strictly for academic research and analysis. It is not designed for, nor licensed to support, commercial or production use, in accordance with the original access conditions of the data sources.

| File Name | GT Top-1 Accuracy | *VOTE-ENSEMBLE* (Ours*) | Extract Verify (Ours*) | SCILEAD |
|---|---|---|---|---|
| 1909.13863v1.pdf | 57.1 | 57.1 | 57.1 | **NA** |
| 1807.11164v1.pdf | 77.2 | 77.2 | **75.4** | **18.56** |
| omniVec_2023.pdf | 92.4 | 92.4 | 92.4 | **NA** |
| 1803.00942v3.pdf | NA | NA | NA | NA |
| 1703.09844v5.pdf | 75 | 75 | 75 | NA |
| 1807.11626v3.pdf | 75.2 | **66** | 75.2 | **76.7** |
| 1807.11459v1.pdf | NA | NA | NA | NA |
| 1807.11254v2.pdf | 77.86 | **NA** | **NA** | **-0.82** |
| 1807.10108v5.pdf | NA | NA | NA | NA |
| 1512.03385v1.pdf | NA | NA | NA | NA |
| 1909.11155v1.pdf | 76.82 | 76.82 | 76.82 | 76.82 |
| 1807.10119v3.pdf | NA | NA | NA | NA |

Table 7: Ground-truth versus extractor outputs on the 12-paper set. Cells in **red bold** indicate a disagreement with the ground-truth.

# I Screenshots from ArXiX Papers with Extract Sections Highlighted

## Abstract

*Designing convolutional neural networks (CNN) for mobile devices is challenging because mobile models need to be small and fast, yet still accurate. Although significant efforts have been dedicated to design and improve mobile CNNs on all dimensions, it is very difficult to manually balance these trade-offs when there are so many architectural possibilities to consider. In this paper, we propose an automated mobile neural architecture search (MNAS) approach, which explicitly incorporate model latency into the main objective so that the search can identify a model that achieves a good trade-off between accuracy and latency. Unlike previous work, where latency is considered via another, often inaccurate proxy (e.g., FLOPS), our approach directly measures real-world inference latency by executing the model on mobile phones. To further strike the right balance between flexibility and search space size, we propose a novel factorized hierarchical search space that encourages layer diversity throughout the network. Experimental results show that our approach consistently outperforms state-of-the-art mobile CNN models across multiple vision tasks. On the ImageNet classification task, our MnasNet achieves 75.2% top-1 accuracy with 78ms latency on a Pixel phone, which is* $1.8\times$ *faster than MobileNetV2 [29] with 0.5% higher accuracy and* $2.3\times$ *faster than NASNet [36] with 1.2% higher accuracy. Our MnasNet also achieves better mAP quality than MobileNets for COCO object detection. Code is at* `https://github.com/tensorflow/tpu/tree/master/models/official/mnasnet`.

Figure 5: **Easy example.** Top-1 accuracy (75.2 %) is stated plainly in the abstract of *1807.11626v3.pdf*.

tational budgets. We plot the performance of each MSDNet as a gray curve; we select the best model for each budget based on its accuracy on the validation set, and plot the corresponding accuracy as a black curve. The plot shows that the predictions of MSDNets with dynamic evaluation are substantially more accurate than those of ResNets and DenseNets that use the same amount of computation. For instance, with an average budget of $1.7 \times 10^9$ FLOPs, MSDNet achieves a top-1 accuracy of $\sim$75%, which is $\sim$6% higher than that achieved by a ResNet with the same number of FLOPs. Compared to the computationally efficient DenseNets, MSDNet uses $\sim 2-3\times$ times fewer

Figure 6: **Easy example.** A single sentence in the main text of *1703.09844v5.pdf* reports top-1 accuracy ( 75 %).

*B. Analysis of network depth in CapsuleNet architecture*

In our study, CapsuleNet shows significantly higher robustness against image degradation than conventional deep CNNs. However, state-of-the-art deep CNNs achieve better recognition accuracy than CapsuleNet for noise-free samples of all datasets. To improve the baseline performance of CapsuleNet, we introduce a novel fusion architecture ***V-CapsNet***

optimize the network by minimizing the marginal loss only. In our experiments, the proposed V-CapsNet fusion architecture achieves 99.83% validation accuracy on the *natural images* dataset, improving the baseline performance of CapsuleNet by 6.2%. Fig. 6 shows the architecture of the proposed V-CapsNet

Figure 7: **Challenging example.** Top-1 accuracy is reported only on a validation split of an ImageNet variant in *1807.10108v5.pdf*.

| method | top-1 err. | top-5 err. |
|---|---|---|
| VGG [41] (ILSVRC'14) | - | 8.43† |
| GoogLeNet [44] (ILSVRC'14) | - | 7.89 |
| VGG [41] (v5) | 24.4 | 7.1 |
| PReLU-net [13] | 21.59 | 5.71 |
| BN-inception [16] | 21.99 | 5.81 |
| ResNet-34 B | 21.84 | 5.71 |
| ResNet-34 C | 21.53 | 5.60 |
| ResNet-50 | 20.74 | 5.25 |
| ResNet-101 | 19.87 | 4.60 |
| ResNet-152 | **19.38** | **4.49** |

Table 4. Error rates (%) of **single-model** results on the ImageNet validation set (except † reported on the test set).

Figure 8: **Challenging example.** The original ResNet paper (*1512.03385.pdf*) only reported on ImageNet validation error rates.

| Dataset | Metric | Modality Encoder | Base Encoder | Modified Encoder | OmniVec (Pre.) | OmniVec (FT) |
|---|---|---|---|---|---|---|
| AudioSet(A) | mAP | AST | 48.5 | 49.4 | 44.7 | 54.8 |
| AudioSet(A+V) | mAP | AST | - | - | 48.6 | 55.2 |
| SSv2 | Top-1 Accuracy | ViViT | 65.4 | 68.6 | 80.1 | 85.4 |
| ImageNet1K | Top-1 Accuracy | ViT | 88.5 | 89.1 | 88.6 | 92.4 |
| Sun RGBD | Top-1 Accuracy | Simple3D-former | 57.3 | 62.4 | 71.4 | 74.6 |

Table 14. **Impact of increasing backbone size of base modality encoders.** All the base modality encoders above are based on ViT architecture. We increase the number of parameters equivalent to our OmniVec-4 model, by replicating the number of layers.

Figure 9: **Challenging example.** In *omniVec_2023.pdf*, the ImageNet Top-1 value (92.4 %) appears as one cell in a table containing multiple datasets.

Table 2. Classification accuracies on ImageNet (ResNet-50)

| Loss Fn. | Parameter | Top-1 | Top-5 |
|---|---|---|---|
| CE | | 76.39 | 93.20 |
| OHEM | $\rho = 0.8$ | 76.27 | 93.21 |
| FL | $\gamma = 0.5$ | 76.72 | 93.06 |
| AL (ours) | $\gamma = 0.5$ | **76.82** | 93.03 |

Figure 10: **Challenging example.** *1909.11155v1.pdf* gives top-1 accuracy only within a table; the metric is absent from the surrounding text.

| Method | shapes | Top-1 acc. | Top-5 acc. |
|---|---|---|---|
| baseline [28] | - | 51.2% | 73.2% |
| **Case 1:** $\alpha$ | $\alpha \in \mathbb{R}^{o\times 1\times 1}$ | 55.5% | 78.5% |
| **Case 2:** $\alpha$ | $\alpha \in \mathbb{R}^{o\times h_{out}\times w_{out}}$ | 56.1% | 79.0% |
| **Case 3:** $\alpha \otimes \beta$ | $\alpha \in \mathbb{R}^{o}, \beta \in \mathbb{R}^{w_{out}\times h_{out}}$ | 56.7% | 79.5% |
| **Case 4:** $\alpha \otimes \beta \otimes \gamma$ | $\alpha \in \mathbb{R}^{o}, \beta \in \mathbb{R}^{w_{out}}$ $\gamma \in \mathbb{R}^{h_{out}}$ | **57.1%** | **79.9%** |

Table 1: Top-1 and Top-5 classification accuracy using a binarized ResNet-18 on Imagenet for various ways of constructing the scaling factor. $\alpha, \beta, \gamma$ are statistically learned via back-propagation. Note that, at test time, all of them can be merged into a single factor, and a single element-wise multiplication is required.

Figure 11: **Challenging example.** *1909.13863v1.pdf* gives top-1 accuracy only within a table.

0.4% accuracy drop. Meanwhile, with the compressed model the inference is accelerated by 2.2×. For *AlexNet* with the ImageNet dataset, we achieve 4.9× model compression at the cost that the top-5 accuracy drops slightly from 81.3% to 80%. For *GoogLeNet* with the ImageNet dataset, the proposed method also brings 2.9× reduction of the model parameters

Figure 12: **Challenging example.** *1807.10119v3.pdf* omits top-1 accuracy, reporting only Top-5 (80 %).

Figure 4 shows performance of CNN features on MIT-indoor dataset. As a baseline we extract CNN features from the entire image (after resizing to 256×256 pixels) and train a multi-class linear SVM. This obtains 72.3% average performance. This is a strong baseline. Razavian et al. (2014) get 58.4% using CNN trained on ImageNet. They improve the result to 69% after data augmentation.

Figure 13: **Challenging example.** *1412.6598v2.pdf* reports multiple references to ImageNet and performance, but no clear top-1 accuracy value.

# Abstract

Training Deep Neural Networks is complicated by the fact that the distribution of each layer's inputs changes during training, as the parameters of the previous layers change. This slows down the training by requiring lower learning rates and careful parameter initialization, and makes it notoriously hard to train models with saturating nonlinearities. We refer to this phenomenon as *internal covariate shift*, and address the problem by normalizing layer inputs. Our method draws its strength from making normalization a part of the model architecture and performing the normalization *for each training mini-batch*. Batch Normalization allows us to use much higher learning rates and be less careful about initialization. It also acts as a regularizer, in some cases eliminating the need for Dropout. Applied to a state-of-the-art image classification model, Batch Normalization achieves the same accuracy with 14 times fewer training steps, and beats the original model by a significant margin. Using an ensemble of batch-normalized networks, we improve upon the best published result on ImageNet classification: reaching 4.9% top-5 validation error (and 4.8% test error), exceeding the accuracy of human raters.

Figure 14: **Challenging example.** *1502.03167v3.pdf* reports only Top-5 validation error (4.9%); no Top-1 value.

## Abstract

*Convolutional Neural Networks demonstrate high performance on ImageNet Large-Scale Visual Recognition Challenges contest. Nevertheless, the published results only show the overall performance for all image classes. There is no further analysis why certain images get worse results and how they could be improved. In this paper, we provide deep performance analysis based on different types of images and point out the weaknesses of convolutional neural networks through experiment. We design a novel multiple paths convolutional neural network, which feeds different versions of images into separated paths to learn more comprehensive features. This model has better presentation for image than the traditional single path model. We acquire better classification results on complex validation set on both top 1 and top 5 scores than the best ILSVRC 2013 classification model.*

Figure 15: **Challenging example.** *1506.04701v3.pdf* with validation results; top-1 accuracy on test set not stated.

| Model | top-1 | top-5 |
|---|---|---|
| DenseNet-121 | 25.02 / 23.61 | 7.71 / 6.66 |
| DenseNet-169 | 23.80 / 22.08 | 6.85 / 5.92 |
| DenseNet-201 | 22.58 / 21.46 | 6.34 / 5.54 |
| DenseNet-264 | 22.15 / 20.80 | 6.12 / 5.29 |

**Table 3:** The top-1 and top-5 error rates on the ImageNet validation set, with single-crop / 10-crop testing.

Figure 16: **Challenging example.** *1608.06993v5.pdf* where Top-1 error rate on ImageNet validation set (e.g., 22.15%) needs conversion to accuracy and split is ambiguous.

Table 1. Classification performance comparison on ImageNet (single crop, single model). VGG-16 and ResNet-152 numbers are only included as a reminder. The version of Inception V3 being benchmarked does not include the auxiliary tower.

| | **Top-1 accuracy** | **Top-5 accuracy** |
|---|---|---|
| **VGG-16** | 0.715 | 0.901 |
| **ResNet-152** | 0.770 | 0.933 |
| **Inception V3** | 0.782 | 0.941 |
| **Xception** | **0.790** | **0.945** |

Figure 17: **Challenging example.** *1610.02357v3.pdf* mentions top-1 accuracy to ImagetNet but split is ambiguous.

***Abstract*— In the field of artificial intelligence, neuromorphic computing has been around for several decades. Deep learning has however made much recent progress such that it consistently outperforms neuromorphic learning algorithms in classification tasks in terms of accuracy. Specifically in the field of image classification, neuromorphic computing has been traditionally using either the temporal or rate code for encoding static images in datasets into spike trains. It is only till recently, that neuromorphic vision sensors are widely used by the neuromorphic research community, and provides an alternative to such encoding methods. Since then, several neuromorphic datasets as obtained by applying such sensors on image datasets (e.g. the neuromorphic CALTECH 101) have been introduced. These data are encoded in spike trains and hence seem ideal for benchmarking of neuromorphic learning algorithms. Specifically, we train a deep learning framework used for image classification on the CALTECH 101 and a collapsed version of the neuromorphic CALTECH 101 datasets. We obtained an accuracy of 91.66% and 78.01% for the CALTECH 101 and neuromorphic CALTECH 101 datasets respectively. For CALTECH 101, our accuracy is close to the best reported accuracy, while for neuromorphic CALTECH 101, it outperforms the last best reported accuracy by over 10%. This raises the question of the suitability of such datasets as benchmarks for neuromorphic learning algorithms.**

Figure 18: Accuracy reported (91.66% and 78.01%) is for CALTECH datasets, not ImageNet.

| | Sources | SYNTH MNIST MNIST-M USPS | SYNTH MNIST SVHN USPS | Avg. |
|---|---|---|---|---|
| | Target | SVHN | MNIST-M | |
| DG | combine sources | 73.2 | 61.9 | 67.5 |
| | MLDG [89] | 68.0 | 65.6 | 66.8 |
| | ADAGE Residual | 68.2 | 65.7 | 66.9 |
| | ADAGE Incremental | **75.8** | **67.0** | **71.4** |
| DA | combine sources | 73.2 | 61.9 | 67.5 |
| | combine DANN [166] | 68.9 | 71.6 | 70.3 |
| | DCTN [166] | 77.5 | 70.9 | 74.2 |
| | ADAGE Residual | 82.3 | 84.1 | 83.2 |
| | ADAGE Incremental | **85.3** | **85.3** | **85.3** |

**Table 3.13.** Classification accuracy results: experiments with 4 sources.

Figure 19: Accuracy values (e.g., 85.3) are shown in table format, but target domains are not ImageNet.

| **Pre-trained Dataset** | IMAGENET-CLS [9, 46] | OPENIMAGES [27] |
|---|---|---|
| CALTECH-256 [15] | **84.7** | 76.7 |
| SUN-397 [53] | **57.3** | 51.1 |
| OXFORD-102 FLOWERS [38] | **87.4** | 83.1 |

Table 6: Linear classification results (Top-1 Accuracy) using `Conv5` features from IMAGENET-CLS and OPENIMAGES pre-trained networks.

Figure 20: Pre-trained models are ImageNet-based, but classification is done on other datasets (e.g., Caltech-256).

| | | Classification | | Localization | |
|---|---|---|---|---|---|
| | | **Top-1** | **Top-5** | **Top-1** | **Top-5** |
| VGG-16 | Backprop [51] | 30.38 | 10.89 | 61.12 | 51.46 |
| | c-MWP [58] | 30.38 | 10.89 | 70.92 | 63.04 |
| | Grad-CAM (ours) | 30.38 | 10.89 | **56.51** | 46.41 |
| | CAM [59] | 33.40 | 12.20 | 57.20 | **45.14** |
| AlexNet | c-MWP [58] | 44.2 | 20.8 | 92.6 | 89.2 |
| | Grad-CAM (ours) | 44.2 | 20.8 | 68.3 | 56.6 |
| GoogleNet | Grad-CAM (ours) | 31.9 | 11.3 | 60.09 | 49.34 |
| | CAM [59] | 31.9 | 11.3 | 60.09 | 49.34 |

Table 1: Classification and localization error % on ILSVRC-15 val (lower is better) for VGG-16, AlexNet and GoogleNet. We see that Grad-CAM achieves superior localization errors without compromising on classification performance.

Figure 21: Classification and localization error rates (%) on ILSVRC-15 validation set from `1610.02391v4.pdf`. The table reports Top-1 classification error for models like VGG-16 and AlexNet. Top-1 metrics on test set is not stated.

TABLE IV: Classification on ImageNet-1k

| Model | Type | Parameters (M) | GMACs | Epochs | Top-1 Accuracy (%) |
|---|---|---|---|---|---|
| ResNet18 [14] | CNN | 11.7 | 1.82 | 300 | 69.7 |
| ResNet50 [14] | CNN | 25.6 | 4.1 | 300 | 80.4 |
| ConvNext-T [70] | CNN | 28.6 | 7.4 | 300 | 82.7 |
| EfficientFormer-L1 [42] | CNN-ViT | 12.3 | 1.3 | 300 | 79.2 |
| EfficientFormer-L3 [42] | CNN-ViT | 31.3 | 3.9 | 300 | 82.4 |
| EfficientFormer-L7 [42] | CNN-ViT | 82.1 | 10.2 | 300 | 83.3 |
| LeViT-192 [71] | CNN-ViT | 10.9 | 0.7 | 1000 | 80.0 |
| LeViT-384 [71] | CNN-ViT | 39.1 | 2.4 | 1000 | 82.6 |
| EfficientFormerV2-S2 [43] | CNN-ViT | 12.6 | 1.3 | 300 | 81.6 |
| EfficientFormerV2-L [43] | CNN-ViT | 26.1 | 2.6 | 300 | 83.3 |
| PVT-Small [72] | ViT | 24.5 | 3.8 | 300 | 79.8 |
| PVT-Large [72] | ViT | 61.4 | 9.8 | 300 | 81.7 |
| DeiT-S [73] | ViT | 22.5 | 4.5 | 300 | 81.2 |
| Swin-T [23] | ViT | 29.0 | 4.5 | 300 | 81.4 |
| PoolFormer-s12 [74] | Pool | 12.0 | 2.0 | 300 | 77.2 |
| PoolFormer-s24 [74] | Pool | 21.0 | 3.6 | 300 | 80.3 |
| PoolFormer-s36 [74] | Pool | 31.0 | 5.2 | 300 | 81.4 |
| PViHGNN-Ti [28] | GNN | 12.3 | 2.3 | 300 | 78.9 |
| PViHGNN-S [28] | GNN | 28.5 | 6.3 | 300 | 82.5 |
| PViHGNN-B [28] | GNN | 94.4 | 18.1 | 300 | 83.9 |
| PViG-Ti [27] | GNN | 10.7 | 1.7 | 300 | 78.2 |
| PViG-S [27] | GNN | 27.3 | 4.6 | 300 | 82.1 |
| PViG-B [27] | GNN | 92.6 | 16.8 | 300 | 83.7 |
| PVG-S [29] | GNN | 22.0 | 5 | 300 | 83.0 |
| PVG-M [29] | GNN | 42.0 | 8.9 | 300 | 83.7 |
| PVG-B [29] | GNN | 79.0 | 16.9 | 300 | 84.2 |
| MobileViG-S [30] | CNN-GNN | 7.2 | 1.0 | 300 | 78.2 |
| MobileViG-M [30] | CNN-GNN | 14.0 | 1.5 | 300 | 80.6 |
| MobileViG-B [30] | CNN-GNN | 26.7 | 2.8 | 300 | 82.6 |
| GreedyViG-S [26] | CNN-GNN | 12.0 | 1.6 | 300 | 81.1 |
| GreedyViG-M [26] | CNN-GNN | 21.9 | 3.2 | 300 | 82.9 |
| GreedyViG-B [26] | CNN-GNN | 30.9 | 5.2 | 300 | 83.9 |
| **CViG-Ti (Ours)** | **CNN-GNN** | **11.5** | **1.3** | **300** | **80.3** |
| **CViG-S (Ours)** | **CNN-GNN** | **28.2** | **4.2** | **300** | **83.7** |
| **CViG-B (Ours)** | **CNN-GNN** | **104.8** | **16.2** | **300** | **85.6** |
| **CViG-B$^{\dagger}$ (Ours)** | **CNN-GNN** | **105.2** | **62.3** | **300** | **87.2** |

Figure 22: Metric table from `2501.10640v2.pdf`. No explicit Top-1 label or split is provided. Extracted value (85.6) does not match the ground truth (87.2).

TABLE VIII

IMAGE CLASSIFICATION PERFORMANCE ON IMAGENET. UNDERLINE INDICATES FLOPS OR METRICS ON PAR WITH THE BASELINE.

| Model | Resolution | #FLOPs | Top-1 Acc |
|---|---|---|---|
| DeiT-B [2] | 224 | 17.2G | 81.8 |
| PIIP-TSB (ours) | 368/192/128 | 17.4G | 82.1 |
| ViT-L [4] | 224 | 61.6G | 84.0 |
| ViT-L [4] (our impl.) | 224 | 61.6G | 85.2 |
| PIIP-SBL (ours) | 320/160/96 | 39.0G | 85.2 |
| PIIP-SBL (ours) | 384/192/128 | 61.2G | 85.9 |

Figure 23: Large benchmark comparison in `2501.07783v1.pdf` with top-1 accuracy buried among multiple datasets. The extracted value of 82.0 does not match the ground truth (85.9).

TABLE 2

Image classification performance (Top-1 Accuracy) on ImageNet-1k under varying input resolutions. FLOPs are measured at input resolution of $224 \times 224$.

| Method | Publication | Param. | FLOPs | Input Resolution | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | $224^2$ | $256^2$ | $384^2$ | $512^2$ | $640^2$ | $768^2$ | $1024^2$ | $1280^2$ | $1408^2$ | $1536^2$ |
| VMamba | NeurIPS'24 | 31M | 4.9G | 82.5 | 82.5 | 82.5 | 81.1 | 79.3 | 76.1 | 62.3 | 50.2 | 45.1 | 40.9 |
| GrootV | NeurIPS'24 | 30M | 4.8G | 83.4 | 83.9 | 83.6 | 82.0 | 80.1 | 77.6 | 67.9 | 52.4 | 45.0 | 39.1 |
| MILA | NeurIPS'24 | 25M | 4.2G | 83.5 | 83.9 | 83.5 | 81.7 | 79.6 | 76.8 | 63.7 | 49.6 | 42.8 | 36.8 |
| MSVMamba | NeurIPS'24 | 33M | 4.6G | 82.8 | 82.5 | 82.3 | 80.9 | 78.8 | 75.1 | 63.0 | 54.9 | 49.6 | 44.0 |
| Spatial Mamba | ICLR'25 | 27M | 4.5G | 83.5 | 83.6 | 83.0 | 80.2 | 77.4 | 74.4 | 66.1 | 53.7 | 46.4 | 38.7 |
| Mamba® | CVPR'25 | 29M | 4.6G | 81.1 | 45.7 | 25.4 | 12.8 | 7.8 | 5.3 | 2.8 | 1.8 | 1.6 | 1.4 |
| MambaVision | CVPR'25 | 32M | 4.4G | 82.3 | 81.7 | 79.8 | 77.6 | 74.8 | 71.2 | 59.6 | 46.4 | 39.7 | 34.5 |
| FractalMamba | AAAI'25 | 31M | 4.8G | 83.0 | 83.5 | 83.9 | 83.0 | 81.8 | 80.3 | 76.3 | 65.9 | 58.8 | 52.1 |
| FractalMamba++ | Year'25 | 30M | 4.8G | 83.0 | 83.3 | 84.1 | 83.9 | 83.0 | 81.9 | 78.8 | 74.3 | 71.3 | 67.5 |
| MSVMamba | NeurIPS'24 | 12M | 1.5G | 79.8 | 80.1 | 80.0 | 78.3 | 75.8 | 72.0 | 59.4 | 43.9 | 36.5 | 29.9 |
| Efficient VMamba | AAAI'25 | 11M | 1.3G | 78.7 | 79.6 | 79.5 | 77.3 | 75.2 | 72.4 | 64.2 | 54.1 | 42.6 | 38.3 |
| FractalMamba++ | Year'25 | 11M | 1.6G | 79.5 | 80.6 | 82.0 | 81.3 | 80.1 | 78.3 | 73.3 | 66.3 | 61.7 | 56.1 |
| MSVMamba | NeurIPS'24 | 7M | 0.9G | 77.3 | 77.7 | 77.4 | 75.0 | 71.7 | 65.8 | 48.0 | 31.0 | 23.8 | 18.3 |
| ViM | ICML'24 | 7M | 1.5G | 76.1 | 76.3 | 70.4 | 67.4 | 51.4 | 30.6 | 16.1 | 7.2 | 4.1 | 1.8 |
| Efficient VMamba | AAAI'25 | 6M | 0.8G | 76.5 | 76.9 | 76.5 | 73.8 | 70.4 | 65.8 | 52.0 | 36.2 | 29.4 | 24.1 |
| FractalMamba++ | Year'25 | 7M | 1.0G | 77.3 | 78.4 | 79.5 | 78.4 | 76.4 | 73.7 | 66.5 | 55.2 | 48.1 | 42.5 |

Figure 24: Ambiguous accuracy reporting in `2505.14062v1.pdf`. Top-1 accuracy for ImageNet is co-listed with CIFAR/Tiny-ImageNet rows. Systems failed to extract a valid value.

Table 1: **Class-wise Bias and Distillation.** The number of statistically significantly affected classes comparing the class-wise accuracy of *teacher vs. Distilled Student (DS) models*, denoted #TC, and *Non-Distilled Student (NDS) vs. distilled student models*, denoted #SC.

| Teacher/Student | | CIFAR-100 | | | | | | ImageNet | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ResNet56/ResNet20 | | | DenseNet169/DenseNet121 | | | ResNet50/ResNet18 | | | ViT-Base/TinyViT | | |
| Model | Temp | Test Acc. (%) | #SC | #TC | Test Acc. (%) | #SC | #TC | Test Top-1 Acc. (%) | #SC | #TC | Test Top-1 Acc. (%) | #SC | #TC |
| Teacher | - | $70.87 \pm 0.21$ | - | - | $72.43 \pm 0.15$ | - | - | $76.1 \pm 0.13$ | - | - | $81.02 \pm 0.07$ | - | - |
| NDS | - | $68.39 \pm 0.17$ | - | - | $70.17 \pm 0.16$ | - | - | $68.64 \pm 0.21$ | - | - | $78.68 \pm 0.19$ | - | - |
| DS | 2 | $68.63 \pm 0.24$ | 5 | 15 | $70.93 \pm 0.21$ | 4 | 12 | $68.93 \pm 0.23$ | 77 | 314 | $78.79 \pm 0.21$ | 83 | 397 |
| DS | 3 | $68.92 \pm 0.21$ | 7 | 12 | $71.08 \pm 0.17$ | 4 | 11 | $69.12 \pm 0.18$ | 113 | 265 | $78.94 \pm 0.14$ | 137 | 318 |
| DS | 4 | $69.18 \pm 0.19$ | 8 | 9 | $71.16 \pm 0.23$ | 5 | 9 | $69.57 \pm 0.26$ | 169 | 237 | $79.12 \pm 0.23$ | 186 | 253 |
| DS | 5 | $69.77 \pm 0.22$ | 9 | 8 | $71.42 \pm 0.18$ | 8 | 9 | $69.85 \pm 0.19$ | 190 | 218 | $79.51 \pm 0.17$ | 215 | 206 |
| DS | 6 | $69.81 \pm 0.15$ | 9 | 8 | $71.39 \pm 0.22$ | 8 | 8 | $69.71 \pm 0.13$ | 212 | 193 | $80.03 \pm 0.19$ | 268 | 184 |
| DS | 7 | $69.38 \pm 0.18$ | 10 | 6 | $71.34 \pm 0.16$ | 9 | 7 | $70.05 \pm 0.18$ | 295 | 174 | $79.62 \pm 0.23$ | 329 | 161 |
| DS | 8 | $69.12 \pm 0.21$ | 13 | 6 | $71.29 \pm 0.13$ | 11 | 7 | $70.28 \pm 0.27$ | 346 | 138 | $79.93 \pm 0.12$ | 365 | 127 |
| DS | 9 | $69.35 \pm 0.27$ | 18 | 9 | $71.51 \pm 0.23$ | 12 | 9 | $70.52 \pm 0.09$ | 371 | 101 | $80.16 \pm 0.17$ | 397 | 96 |
| DS | 10 | $69.24 \pm 0.19$ | 22 | 11 | $71.16 \pm 0.21$ | 14 | 10 | $70.83 \pm 0.15$ | 408 | 86 | $79.98 \pm 0.12$ | 426 | 78 |

Figure 25: In `2410.08407v2.pdf`, Top-1 accuracy (81.02) appears in a table with closely related numbers. Extractors returned 76.1 or 79.51, misaligned with the correct value.

Table 2: Top-1 and Top-5 classification accuracy (%) on ImageNet. † denotes the results from [18] and ‡ from [43]. The best results are highlighted in **Bold**.

| Model + Method | Top-1 / Top-5 |
|---|---|
| ResNet50 + Hard Label | 76.30 / 93.05 |
| ResNet50 + LS[34] | 76.67 / -‡ |
| ResNet50 + CutOut[44] | 77.07 / 93.34† |
| ResNet50 + Disturb Label[35] | 76.41 / 93.10† |
| ResNet50 + BYOT[8] | 76.96 / 93.49† |
| ResNet50 + TF-KD[7] | 76.56 / -‡ |
| ResNet50 + CS-KD[21] | 76.78 / -‡ |
| ResNet50 + Zipf's LS[43] | 77.25 /-‡ |
| ResNet152 (Teacher)<br>ResNet50 + KD[1] | 77.49 /- |
| ResNet50 + Ours (2×2) | **77.85** / **93.57**<br>(1.55↑) / (0.52↑) |
| ResNet50 + Ours (4×4) | **77.59** / **93.56**<br>(1.29↑) / (0.51↑) |
| MobileNetV2 [41] | 60.05 / 83.20 |
| MobileNetV2 + Ours (2×2) | **60.83** / **84.31**<br>(0.78↑) / (1.11↑) |
| MViTv2 [42] | 77.71 / - |
| MViTv2 + Ours (2×2) | **80.99** / -<br>(3.28↑) / - |

Figure 26: Example from `2505.14124v1.pdf` where 80.99 is reported, but the Top-1 metric is embedded among unlabeled entries. Systems extracted 77.85, a nearby but incorrect value.

| Dataset | Approach | ResNet-18 | | Swin-T | | MobileNet-V2 | | VGG-16bn | |
|---|---|---|---|---|---|---|---|---|---|
| | | top-1 | Rem. | top-1 | Rem. | top-1 | Rem. | top-1 | Rem. |
| CIFAR-10 | Dense model | 92.00 | 0/17 | 91.63 | 0/12 | 93.64 | 0/35 | 93.09 | 0/15 |
| | Smallest weights | 88.49 | 11/17 | 86.92 | 3/12 | 10.00 | 1/35 | 90.53 | 7/15 |
| | Smallest gradients | 88.60 | 11/17 | 86.96 | 3/12 | 10.00 | 1/35 | 90.4 | 7/15 |
| | EGP | 90.64 | 5/17 | 86.04 | 6/12 | 92.22 | 6/35 | 10.00 | 1/15 |
| | LF | 90.65 | 1/17 | 85.73 | 2/12 | 89.24 | 9/35 | 86.46 | 1/15 |
| | EASIER | 86.53 | 11/17 | 91.25 | 6/12 | 92.45 | 16/35 | 93.03 | 7/15 |
| | TLC | **$90.91 \pm 0.57$** | **12/17** | **$91.98 \pm 0.07$** | **6/12** | **$92.97 \pm 0.38$** | **17/35** | **$93.61 \pm 0.23$** | **7/15** |
| Tiny-Inet | Dense model | 41.86 | 0/17 | 75.88 | 0/12 | 45.70 | 0/35 | 58.44 | 0/15 |
| | Smallest weights | 37.42 | 8/17 | 72.90 | 1/12 | 0.5 | 1/35 | 56.88 | 1/15 |
| | Smallest gradients | 37.88 | 8/17 | 72.92 | 1/12 | 0.5 | 1/35 | 57.34 | 1/15 |
| | LF | 37.86 | 4/17 | 50.54 | 1/12 | 25.88 | 12/35 | 31.22 | 1/15 |
| | EGP | 37.44 | 5/17 | 71.48 | 1/12 | 46.88 | 1/35 | — | — |
| | EASIER | 35.84 | 6/17 | 70.94 | 1/12 | 47.58 | 11/35 | 55.16 | 1/15 |
| | TLC | **$38.66 \pm 0.68$** | **9/17** | **$74.07 \pm 0.02$** | **1/12** | **$47.84 \pm 0.55$** | **16/35** | **$57.63 \pm 0.65$** | 1/15 |
| PACS | Dense model | 79.70 | 0/17 | 97.00 | 0/12 | 96.10 | 0/35 | 96.10 | 0/15 |
| | Smallest weights | 84.30 | 8/17 | 95.10 | 3/12 | 18.50 | 1/35 | 95.20 | 3/15 |
| | Smallest gradients | 83.60 | 6/17 | 95.90 | 3/12 | 18.50 | 1/35 | 95.50 | 1/15 |
| | LF | 82,90 | 3/17 | 87,70 | 2/12 | 79.70 | 1/35 | 93.60 | 1/15 |
| | EGP | 81.60 | 3/17 | 93.50 | 4/12 | 17.70 | 3/35 | — | — |
| | EASIER | **88.30** | **9/17** | 93.80 | 3/12 | 94.40 | 7/35 | 95.20 | 3/15 |
| | TLC | $84.80 \pm 0.78$ | 9/17 | **$96.57 \pm 0.41$** | **4/12** | **$94.87 \pm 0.19$** | **11/35** | **$95.98 \pm 0.22$** | **4/15** |
| VLCS | Dense model | 67.85 | 0/17 | 85.83 | 0/12 | 81.83 | 0/35 | 84.62 | 0/15 |
| | Smallest weights | 65.89 | 16/17 | 69.99 | 5/12 | 6.43 | 1/35 | 80.71 | 7/15 |
| | Smallest gradients | 66.26 | 11/17 | 70.18 | 5/12 | 6.43 | 1/35 | 80.99 | 7/15 |
| | LF | 63.28 | 7/17 | 70.92 | 1/12 | 68.87 | 2/35 | 80.24 | 2/15 |
| | EGP | 64.40 | 5/17 | 82.76 | 3/12 | 45.85 | 2/35 | — | — |
| | EASIER | 54.24 | 15/17 | 78.19 | 5/12 | 72.88 | 22/35 | 78.84 | 6/15 |
| | TLC | **$66.43 \pm 0.66$** | **16/17** | **$82.79 \pm 0.31$** | **5/12** | **$76.11 \pm 1.18$** | **23/35** | **$81.41 \pm 0.42$** | **7/15** |
| ImageNet | Dense model | 68.28 | 0/17 | 81.08 | 0/12 | 71.87 | 0/35 | 73.37 | 0/15 |
| | Smallest weights | 67.80 | 2/17 | 79.74 | 1/12 | 0.1 | 1/35 | 70.67 | 1/15 |
| | Smallest gradients | 67.56 | 2/17 | 79.71 | 1/12 | 0.1 | 1/35 | 70.12 | 1/15 |
| | LF | 67.62 | 1/17 | 73.51 | 1/12 | 7.89 | 1/35 | 72.22 | 2/15 |
| | EGP | 61.73 | 2/17 | 78.62 | 1/12 | 0.1 | 1/35 | — | — |
| | EASIER | 67.20 | 2/17 | 78.78 | 1/12 | 41.14 | 2/35 | 1.19 | 1/15 |
| | TLC | **67.81** | **2/17** | **79.96** | **1/12** | **59.43** | **2/35** | **72.89** | **2/15** |

Table 1: Test performance (top-1) and the number of removed layers (Rem.) for all image classification setups considered. The best results between Smallest weights/gradients, LF, EGP, EASIER, and TLC are in **bold**.

Figure 27: Figure from `2412.15077v1.pdf`, where top-1 accuracy (79.96) appears in a multi-column architecture table. No extractor returned the correct value.

| Method | Venue | Input Size | Epochs | Token Mixer | **Throughput** (im/s) ↑ | **Thr**$_{\text{rel}}$ ↑ | **Latency** (ms) ↓ | **Top-1** (%) ↑ | Params (M) | FLOPs (M) |
|---|---|---|---|---|---|---|---|---|---|---|
| MobileViTV2 0.5 [45] | Arxiv 2022 | $256^2$ | 300 | Att. | 6,702 | ×0.32 | 0.149 | 70.2 | 1.4 | 466 |
| MobileOne-S0 [67] | CVPR 2023 | $224^2$ | 300 | Conv | 13,313 | ×0.64 | 0.075 | 71.4 | 2.1 | 275 |
| EMO-1M [83] | ICCV 2023 | $224^2$ | 300 | Att. | 6,945 | ×0.34 | 0.144 | 71.5 | 1.3 | 261 |
| MobileFormer-96M [3] | CVPR 2022 | $224^2$ | 450 | Att. | 11,554 | ×0.56 | 0.087 | 72.8 | 4.6 | 96 |
| SHViT-S1 [80] | CVPR 2024 | $224^2$ | 300 | Att. | 19,868 | ×0.96 | 0.050 | 72.8 | 6.3 | 241 |
| **EfficientViM-M1** | - | $224^2$ | 300 | SSD | **20,731** | **×1.00** | **0.048** | 72.9 | 6.7 | 239 |
| MobileNetV3-L 0.75 [22] | ICCV 2019 | $224^2$ | 600 | Conv | 10,846 | ×0.52 | 0.092 | 73.3 | 4.0 | 155 |
| EfficientViT-M3 [36] | CVPR 2023 | $224^2$ | 300 | Att. | 16,045 | ×0.77 | 0.062 | 73.4 | 6.9 | 263 |
| **EfficientViM-M1** | - | $224^2$ | 450 | SSD | **20,731** | **×1.00** | **0.048** | **73.5** | 6.7 | 239 |
| EfficientFormerV2-S0 [33] | NeurIPS 2022 | $224^2$ | 300 | Att. | 1,350 | ×0.08 | 0.741 | 73.7 | 3.5 | 407 |
| EfficientViT-M4 [36] | CVPR 2023 | $224^2$ | 300 | Att. | 15,807 | ×0.93 | 0.063 | 74.3 | 8.8 | 299 |
| EdgeViT-XXS [48] | ECCV 2022 | $224^2$ | 300 | Att. | 5,990 | ×0.35 | 0.167 | 74.4 | 4.1 | 556 |
| EMO-2M [83] | ICCV 2023 | $224^2$ | 300 | Att. | 4,990 | ×0.29 | 0.200 | 75.1 | 2.3 | 439 |
| MobileNetV3-L 1.0 [22] | ICCV 2019 | $224^2$ | 600 | Conv | 9,493 | ×0.56 | 0.105 | 75.2 | 5.4 | 217 |
| MobileFormer-151M [3] | CVPR 2022 | $224^2$ | 450 | Att. | 8,890 | ×0.52 | 0.112 | 75.2 | 7.6 | 151 |
| SHViT-S2 [80] | CVPR 2024 | $224^2$ | 300 | Att. | 15,899 | ×0.93 | 0.063 | 75.2 | 11.4 | 366 |
| **EfficientViM-M2** | - | $224^2$ | 300 | SSD | **17,005** | **×1.00** | **0.059** | 75.4 | 13.9 | 355 |
| MobileViTV2 0.75 [45] | Arxiv 2022 | $256^2$ | 300 | Att. | 4,409 | ×0.26 | 0.227 | 75.6 | 2.9 | 1030 |
| FastViT-T8 [66] | ICCV 2023 | $256^2$ | 300 | Att. | 4,365 | ×0.26 | 0.229 | 75.6 | 3.6 | 705 |
| **EfficientViM-M2** | - | $224^2$ | 450 | SSD | **17,005** | **×1.00** | **0.059** | **75.8** | 13.9 | 355 |
| EfficientMod-XXS [43] | ICLR 2024 | $224^2$ | 300 | Att. | 7022 | ×0.59 | 0.142 | 76.0 | 4.7 | 583 |
| ConvNeXtV2-A [72] | CVPR 2023 | $224^2$ | 300 | Conv | 7563 | ×0.63 | 0.132 | 76.2 | 3.7 | 552 |
| EfficientViT-M5 [36] | CVPR 2023 | $224^2$ | 300 | Att. | 11,105 | ×0.93 | 0.090 | 77.1 | 12.4 | 522 |
| MobileOne-S2 [67] | CVPR 2023 | $224^2$ | 300 | Conv | 5,360 | ×0.45 | 0.187 | 77.4 | 7.8 | 1299 |
| SHViT-S3 [80] | CVPR 2024 | $224^2$ | 300 | Att. | 11,873 | ×0.99 | 0.084 | 77.4 | 14.2 | 601 |
| EdgeViT-XS [48] | ECCV 2022 | $224^2$ | 300 | Att. | 4,405 | ×0.37 | 0.227 | 77.5 | 6.7 | 1136 |
| **EfficientViM-M3** | - | $224^2$ | 300 | SSD | **11,952** | **×1.00** | **0.084** | 77.6 | 16.6 | 656 |
| MobileFormer-294M [3] | CVPR 2022 | $224^2$ | 450 | Att. | 6,576 | ×0.55 | 0.152 | 77.9 | 11.4 | 294 |
| EfficientFormerV2-S1 [33] | NeurIPS 2022 | $224^2$ | 300 | Att. | 1,248 | ×0.10 | 0.801 | 77.9 | 6.1 | 668 |
| **EfficientViM-M3** | - | $224^2$ | 450 | SSD | **11,952** | **×1.00** | **0.084** | **77.9** | 16.6 | 656 |
| ConvNeXtV2-F [72] | CVPR 2023 | $224^2$ | 300 | Conv | 6,405 | ×0.78 | 0.156 | 78.0 | 5.2 | 785 |
| MobileViTV2 1.0 [45] | Arxiv 2022 | $256^2$ | 300 | Att. | 2,977 | ×0.36 | 0.336 | 78.1 | 4.9 | 1844 |
| MobileOne-S3 [67] | CVPR 2023 | $224^2$ | 300 | Conv | 4,181 | ×0.51 | 0.239 | 78.1 | 10.1 | 1896 |
| EfficientMod-XS [43] | ICLR 2024 | $224^2$ | 300 | Att. | 5,321 | ×0.65 | 0.188 | 78.3 | 6.6 | 778 |
| EMO-6M [83] | ICCV 2023 | $224^2$ | 300 | Att. | 3,266 | ×0.40 | 0.306 | 79.0 | 6.1 | 961 |
| FastViT-T12 [66] | ICCV 2023 | $256^2$ | 300 | Att. | 2,741 | ×0.34 | 0.365 | 79.1 | 6.8 | 1419 |
| MobileFormer-508M [3] | CVPR 2022 | $224^2$ | 450 | Att. | 4,586 | ×0.56 | 0.218 | 79.3 | 14.0 | 508 |
| MobileOne-S4 [67] | CVPR 2023 | $224^2$ | 300 | Conv | 3,041 | ×0.37 | 0.329 | 79.4 | 14.8 | 2978 |
| SHViT-S4 [80] | CVPR 2024 | $256^2$ | 300 | Att. | 8,024 | ×0.98 | 0.124 | 79.4 | 16.5 | 986 |
| **EfficientViM-M4** | - | $256^2$ | 300 | SSD | **8,170** | **×1.00** | **0.122** | 79.4 | 19.6 | 1111 |
| MobileViTV2 1.25 [45] | Arxiv 2022 | $256^2$ | 300 | Att. | 2,409 | ×0.24 | 0.415 | 79.6 | 7.5 | 2857 |
| **EfficientViM-M4** | - | $256^2$ | 450 | SSD | **8,170** | **×1.00** | **0.122** | **79.6** | 19.6 | 1111 |

Table 3. **Comparison of efficient networks on ImageNet-1K [10] classification.** Results are sorted by accuracy. We also denote the relative throughput **Thr**$_{\text{rel}}$ of each method compared to EfficientViM in each split.

Figure 28: In `2411.15241v1.pdf`, 79.6 is reported in a dense comparison table with ViT variants. Systems hallucinated or extracted higher values (e.g., 82.0, 84.2).

TABLE 2: Top-1 and Top-5 accuracies comparison on ImageNet-1K using ResNet-50, on Tiny ImageNet-200 and CIFAR-100 using PreActResNet-18. FGSM error rates on CIFAR-100 and Tiny-ImageNet-200 datasets are also computed for PreActResNet-18. Compared numbers are taken either from the original papers or from Kang and Kim (2023), following the exact protocols.

| Method | ImageNet-1K | | Tiny ImageNet-200 | | | CIFAR-100 | | |
|---|---|---|---|---|---|---|---|---|
| | Top-1 Acc (%) | Top-5 Acc (%) | Top-1 Acc (%) | Top-5 Acc (%) | FGSM Error (%) | Top-1 Acc (%) | Top-5 Acc (%) | FGSM Error (%) |
| Vanilla-He et al. (2016b) | 75.97 | 92.66 | 57.23 | 73.65 | 42.77 | 76.33 | 91.02 | 23.67 |
| AugMix Hendrycks et al. (2019) | 76.75 | 93.30 | 55.97 | 74.68 | - | 75.31 | 91.62 | 43.33 |
| ManifoldMix Verma et al. (2019) | 76.85 | 93.50 | 58.01 | 74.12 | 41.99 | 79.02 | 93.37 | 20.98 |
| Mixup Zhang et al. (2018) | 77.03 | 93.52 | 56.59 | 73.02 | 43.41 | 76.84 | 92.42 | 23.16 |
| CutMix Yun et al. (2019b) | 77.08 | 93.45 | 56.67 | 75.52 | 43.33 | 76.80 | 91.91 | 23.20 |
| Guided-SR Kim et al. (2020a) | 77.20 | 93.66 | 55.97 | 74.68 | - | 80.60 | 94.00 | - |
| PixMix Hendrycks et al. (2022) | 77.40 | - | - | - | - | 79.70 | - | - |
| PuzzleMix Kim et al. (2020a) | 77.51 | 93.76 | 63.48 | 75.52 | 36.52 | 80.38 | 94.15 | 19.62 |
| GuidedMix Kang and Kim (2023) | 77.53 | 93.86 | 64.63 | 82.49 | - | 81.20 | 94.88 | - |
| Co-Mixup Kim et al. (2020b) | 77.63 | 93.84 | 64.15 | - | - | 80.15 | - | - |
| YOCO Han et al. (2022) | 77.88 | - | - | - | - | - | - | - |
| Azizi et al. Azizi et al. (2023) | 78.17 | - | - | - | - | - | - | - |
| **GenMix** | **78.73** | **95.47** | **65.80** | **83.70** | **34.47** | **82.58** | **95.51** | **16.83** |

Figure 29: Metric table from `2412.02366v3.pdf`, where 78.73 is buried among comparisons across variants. Extractors returned mismatched outputs like 73.3 or 65.8.

Table 2. ImageNet-1k results for HgVT and other isotropic networks. ❋ CNN, ♦Transformer, ★GNN, ■HGNN, and ▲HgVT.

| Model | Params | FLOPs | ImNet Top-1 | ReaL Top-1 | V2 Top-1 |
|---|---|---|---|---|---|
| ❋ ResMLP-S12 conv3x3 [52] | 16.7M | 3.2B | 77.0 | 84.0 | 65.5 |
| ❋ ConvMixer-768/32 [55] | 21.1M | 20.9B | 80.2 | – | – |
| ❋ ConvMixer-1536/20 [55] | 51.6M | 51.1B | 81.4 | – | – |
| ♦DINOv1-S [2] | 21.7M | 4.6B | 77.0 | – | – |
| ♦ViT-B/16 [12] | 86.4M | 55.5B | 77.9 | 83.6 | – |
| ♦DeiT-Ti [53] | 5.7M | 1.3B | 72.2 | 80.1 | 60.4 |
| ♦DeiT-S [53] | 22.1M | 4.6B | 79.8 | 85.7 | 68.5 |
| ♦DeiT-B [53] | 86.4M | 17.6B | 81.8 | 86.7 | 71.5 |
| ★ViG-Ti [16] | 7.1M | 1.3B | 73.9 | – | – |
| ★ViG-S [16] | 22.7M | 4.5B | 80.4 | – | – |
| ★ViG-B [16] | 86.8M | 17.7B | 82.3 | – | – |
| ■ViHGNN-Ti [17] | 8.2M | 1.8B | 74.3 | – | – |
| ■ViHGNN-S [17] | 23.2M | 5.6B | 81.5 | – | – |
| ■ViHGNN-B [17] | 88.1M | 19.4B | 82.9 | – | – |
| ▲HgVT-Ti (ours) | 7.7M | 1.8B | **76.2** | **83.2** | **64.3** |
| ▲HgVT-S (ours) | 22.9M | 5.5B | 81.2 | **86.7** | **70.1** |

Tab. 2 presents the ImageNet-1k top-1 accuracy of HgVT,

Figure 30: Top-1 accuracy of 76.2 from `2504.08710v1.pdf` is presented in a large table with no metric labels. Extracted values (e.g., 72.2) reflect misalignment.

## 3 Results

We evaluate the baseline IJEPA and our proposed encoder conditioned variant EC-IJEPA on several visual benchmarks consistent with prior work [14, 16]. We follow the setup from Assran et al. [14] to pretrain the baseline IJEPA and our proposed EC-IJEPA on the ImageNet-1k (IN-1k) dataset [13] (see Appendix A for more details). The pretrained encoders are then used to extract representations, by average pooling the output sequence of patch-level tokens from the

Table 1: Classification performance comparison on IN-1k dataset.

| Model | Accuracy |
|---|---|
| IJEPA (ViT-L/16) | 74.8 |
| EC-IJEPA (ViT-L/16) | **76.7** |
| IJEPA (ViT-H/14) | 77.4 |
| EC-IJEPA (ViT-H/14) | **78.1** |

Figure 31: In `2410.10773v1.pdf`, top-1 accuracy (78.1) appears with multiple candidate rows and no clear indicator. Systems returned incorrect values such as 70.0.

| Models | RN50 | RN101 | V-B/16 | V-B/32 |
|---|---|---|---|---|
| Zero-shot CLIP (Radford et al., 2021) | 60.33 | 62.53 | 68.73 | 63.80 |
| CoOp (Gu et al., 2021) | 62.95 | 66.60 | 71.92 | 66.85 |
| CLIP-Adapter (Gao et al., 2024) | 63.59 | 65.39 | 71.13 | 66.19 |
| Tip-Adapter (Zhang et al., 2021) | 62.03 | 64.78 | 70.75 | 65.61 |
| Tip-Adapter-F (Zhang et al., 2021) | 65.51 | 68.56 | 73.69 | 68.65 |
| **CMM** | **66.17** | **68.93** | **74.23** | **69.17** |

Table 2: Comparison of Top-1 accuracy across various methods on the ImageNet dataset using 16-shot learning with different architectures, where 'RN' represents ResNet and 'V-' represents ViT (Dosovitskiy, 2020).

Figure 32: Large table showing top-1 accuracies across various architectures on ImageNet, but split (validation/test) is not explicitly stated.

| | ImageNet | Caltech101 | OxfordPets | StanfordCars | Flowers102 | Food101 | FGVCAircraft | SUN397 | DTD | UCF101 | ImageNetv2 | ImageNet-R | ImageNet-S | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CLIP-S | 73.5 | 94.3 | 93.1 | 76.9 | 76.2 | 90.3 | 30.0 | 67.6 | 52.5 | 73.8 | 60.9 | 74.0 | 46.2 | 69.9 |
| CLIP-DS | 75.5 | 93.7 | 93.5 | 78.1 | 79.5 | 90.9 | 31.8 | 69.0 | 54.8 | 76.2 | 61.9 | 77.7 | 48.8 | 71.6 |
| CuPL | 76.7 | 93.5 | 93.8 | 77.6 | 79.7 | 93.4 | 36.1 | 73.3 | 61.7 | 78.4 | 63.4 | - | - | 75.2 |
| D-CLIP | 75.1 | **97.0** | 93.0 | 75.1 | 79.5 | 91.1 | 31.8 | 69.6 | 56.1 | 76.2 | 62.2 | 76.5 | 48.9 | 71.7 |
| Waffle | 75.1 | 96.2 | 93.2 | 76.5 | 78.3 | 91.5 | 32.5 | 69.4 | 55.3 | 76.0 | 62.3 | 77.0 | 49.1 | 71.7 |
| MPVR (Mix) | 75.9 | 95.4 | 93.1 | 70.6 | 83.8 | 91.4 | 37.6 | 72.5 | 61.6 | 75.8 | 62.2 | 78.4 | 49.7 | 72.9 |
| MPVR (GPT) | **76.8** | 96.1 | 93.7 | 78.3 | 83.6 | 91.5 | 34.4 | 73.0 | 62.9 | 78.1 | 63.4 | 78.2 | 50.6 | 73.9 |
| **Ours (SLAC)** | 73.8 | 96.6 | 96.5 | 88.7 | 77.7 | 92.9 | 65.6 | 73.5 | 58.5 | 85.2 | 67.9 | 89.9 | 66.1 | 79.4 |
| **Ours (TLAC)** | 74.1 | **97.0** | **97.1** | **90.2** | **85.7** | **94.4** | **79.4** | **79.0** | **72.6** | **89.5** | **69.2** | **90.8** | **68.2** | **83.6** |

Table 1. Table compares the results of our models with those of previous training-free methods. Results of previous state-of-the-art models have been taken from [25].The best result is displayed in bold, while the second-highest result is shown in blue. Higher scores represent superior performance.

Figure 33: Top-1 accuracy of 74.1 appears in a multi-dataset benchmark.

| Setting | | $\mathcal{T}$ | $\mathcal{S}$ | Logit | | | | | | Feature | | | | | Logit + Feature | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | KD [2] | DKD [3] | NKD [4] | CTKD [54] | WTTM [5] | WKD-L (ours) | FitNet [24] | CRD [25] | Review -KD [29] | CAT [55] | WKD-F (ours) | CRD+ KD [25] | DPK [7] | FCFD [8] | KD-Zero [56] | WKD-L+ WKD-F (ours) |
| (a) | Top-1 | 73.31 | 69.75 | 71.03 | 71.70 | 71.96 | 71.51 | 72.19 | **72.49** | 70.53 | 71.17 | 71.61 | 71.26 | **72.50** | 71.38 | 72.51 | 72.25 | 72.17 | **72.76** |
| | Top-5 | 91.42 | 89.07 | 90.05 | 90.41 | – | 90.47 | – | **90.75** | 89.87 | 90.13 | 90.51 | 90.45 | **91.00** | 90.49 | 90.77 | 90.71 | 90.46 | **91.08** |
| (b) | Top-1 | 76.16 | 68.87 | 70.50 | 72.05 | 72.58 | – | 73.09 | **73.17** | 70.26 | 71.37 | 72.56 | 72.24 | **73.12** | – | 73.26 | 73.26 | 73.02 | **73.69** |
| | Top-5 | 92.86 | 88.76 | 89.80 | 91.05 | – | – | – | **91.32** | 90.14 | 90.41 | 91.00 | 91.13 | **91.39** | – | 91.17 | 91.24 | 91.05 | **91.63** |

Table 4: Image classification results (Acc, %) on ImageNet. In setting (a), the teacher ($\mathcal{T}$) and student ($\mathcal{S}$) are ResNet34 and ResNet18, respectively, while setting (b) consists of a teacher of ResNet50 and a student of MobileNetV1. We refer to *Table 10 in Section C.4 for additional comparison to competitors with different setups*.

Figure 34: Table 4 compares classification accuracy (%) across methods and settings on ImageNet.

| Source of $\mathcal{P}$ | Description Assignment | Max #desc. ↓ | ImageNet | ImageNetV2 | CUB200 | EuroSAT | Places365 | DTD | Flowers102 |
|---|---|---|---|---|---|---|---|---|---|
| DCLIP | LLM (global eval) | 13 | 61.99 | 55.09 | 51.79 | 43.31 | 39.91 | 43.09 | 62.97 |
| DCLIP | LLM (local-k eval) | 13 | 61.99 | 55.06 | 51.83 | 43.29 | 39.87 | 43.09 | 62.86 |
| DCLIP | *Ours* | 5 | 62.57 | **55.48** | **53.80** | **49.89** | 42.64 | **47.23** | 66.37 |
| Random | *Ours* | 5 | 62.18 | 55.22 | 52.31 | 40.82 | 40.44 | 44.73 | 66.12 |
| Contrastive | LLM | 40 | 62.03 | 54.88 | 52.24 | 46.97 | 40.37 | 44.41 | 62.90 |
| Contrastive | *Ours* | 5 | **62.78** | **55.48** | 53.45 | 49.47 | **42.65** | 46.97 | **67.07** |

Table 1: Image classification in classname-free setup with different assignments and pools. Our method consistently produces the highest accuracies in this setting. We use the best-performing $w_{cls}$ of the respective assignment to ensure a fair comparison.

Figure 35: Top-1 accuracy reported on ImageNet appears in a wide comparison table (e.g., 62.78).