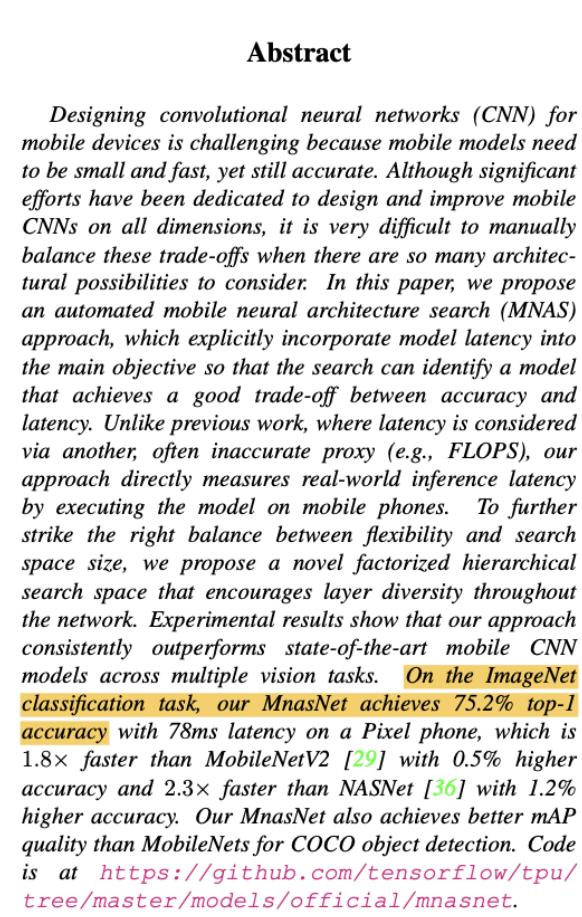
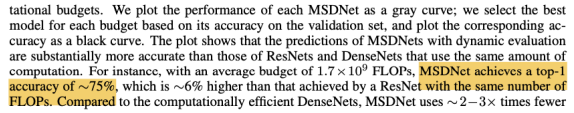


Abstract

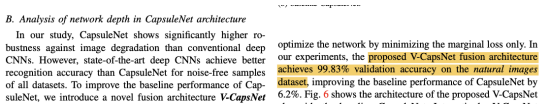
Designing convolutional neural networks (CNN) for mobile devices is challenging because mobile models need to be small and fast, yet still accurate. Although significant efforts have been dedicated to design and improve mobile CNNs on all dimensions, it is very difficult to manually balance these trade-offs 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× faster than MobileNetV2 [29] with 0.5% higher accuracy and 2.3× 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 6: **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 1.7×10^9 FLOPs, MSDNet achieves a top-1 accuracy of ∼75%, which is ∼6% higher than that achieved by a ResNet with the same number of FLOPs. Compared to the computationally efficient DenseNets, MSDNet uses ∼2−3× times fewer

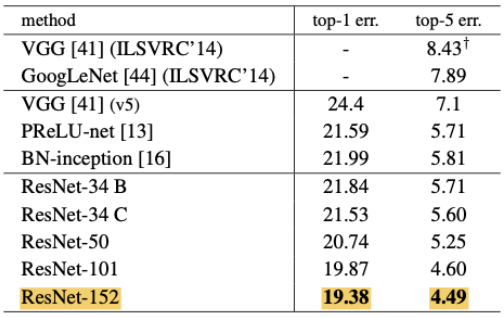
Figure 7: **Easy example.** A single sentence in the main text of *1703.09844v5.pdf* reports top-1 accuracy (75 %).



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

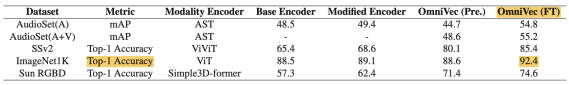
Figure 8: **Challenging example.** Top-1 accuracy is reported only on a validation split of an ImageNet variant in *1807.10108v5.pdf*.



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 9: **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

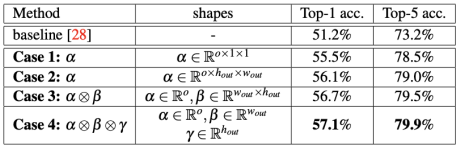
Table 14. **Impact of increasing backbone size of base modality encoders.** All the base modality encoders above are based on ViT architecture. We increase the number of parameters equivalent to our OmniVec-4 model, by replicating the number of layers.

Figure 10: **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 11: **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: α	$\alpha \in \mathbb{R}^{o \times 1 \times 1}$	55.5%	78.5%
Case 2: α	$\alpha \in \mathbb{R}^{o \times h_{out} \times w_{out}}$	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. α, β, γ 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 12: **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 13: **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 14: **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 15: **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 16: **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 17: **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 18: **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 19: 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 20: 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 21: 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 22: 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† (Ours)	**CNN-GNN**	**105.2**	**62.3**	**300**	**87.2**

Figure 23: 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 24: 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 × 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 25: 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 ± 0.21	-	-	72.43 ± 0.15	-	-	76.1 ± 0.13	-	-	81.02 ± 0.07	-	-
NDS	-	68.39 ± 0.17	-	-	70.17 ± 0.16	-	-	68.64 ± 0.21	-	-	78.68 ± 0.19	-	-
DS	2	68.63 ± 0.24	5	15	70.93 ± 0.21	4	12	68.93 ± 0.23	77	314	78.79 ± 0.21	83	397
DS	3	68.92 ± 0.21	7	12	71.08 ± 0.17	4	11	69.12 ± 0.18	113	265	78.94 ± 0.14	137	318
DS	4	69.18 ± 0.19	8	9	71.16 ± 0.23	5	9	69.57 ± 0.26	169	237	79.12 ± 0.23	186	253
DS	5	69.77 ± 0.22	9	8	71.42 ± 0.18	8	9	69.85 ± 0.19	190	218	79.51 ± 0.17	215	206
DS	6	69.81 ± 0.15	9	8	71.39 ± 0.22	8	8	69.71 ± 0.13	212	193	80.03 ± 0.19	268	184
DS	7	69.38 ± 0.18	10	6	71.34 ± 0.16	9	7	70.05 ± 0.18	295	174	79.62 ± 0.23	329	161
DS	8	69.12 ± 0.21	13	6	71.29 ± 0.13	11	7	70.28 ± 0.27	346	138	79.93 ± 0.12	365	127
DS	9	69.35 ± 0.27	18	9	71.51 ± 0.23	12	9	70.52 ± 0.09	371	101	80.16 ± 0.17	397	96
DS	10	69.24 ± 0.19	22	11	71.16 ± 0.21	14	10	70.83 ± 0.15	408	86	79.98 ± 0.12	426	78

Figure 26: 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) ResNet50 + KD[1]	77.49 /-
ResNet50 + Ours (2×2)	**77.85 / 93.57** (1.55↑) / (0.52↑)
ResNet50 + Ours (4×4)	**77.59 / 93.56** (1.29↑) / (0.51↑)
MobileNetV2 [41]	60.05 / 83.20
MobileNetV2 + Ours (2×2)	**60.83 / 84.31** (0.78↑) / (1.11↑)
MViTv2 [42]	77.71 / -
MViTv2 + Ours (2×2)	**80.99** / - (3.28↑) / -

Figure 27: 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 ± 0.57**	**12/17**	**91.98 ± 0.07**	**6/12**	**92.97 ± 0.38**	**17/35**	**93.61 ± 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 ± 0.68**	**9/17**	**74.07 ± 0.02**	**1/12**	**47.84 ± 0.55**	**16/35**	**57.63 ± 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 ± 0.78	9/17	**96.57 ± 0.41**	**4/12**	**94.87 ± 0.19**	**11/35**	**95.98 ± 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 ± 0.66**	**16/17**	**82.79 ± 0.31**	**5/12**	**76.11 ± 1.18**	**23/35**	**81.41 ± 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	1/17	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 28: 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) ↑	Throughput $\mathbf{Thr}_{\text{rel}}$ ↑	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 $\mathbf{Thr}_{\text{rel}}$ of each method compared to EfficientViM in each split.

Figure 29: 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	94.30	-
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 30: 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 31: 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 32: 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 33: 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	ImageNet-v2	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 34: 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 35: 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 36: Top-1 accuracy reported on ImageNet appears in a wide comparison table (e.g., 62.78).