AIO-P: Expanding Neural Performance Predictors Beyond Image Classification

Keith G. Mills[1], Di Niu[1], Mohammad Salameh[2], Weichen Qiu[1], Fred X. Han[2], Puyuan Liu[2], Jialin Zhang[3], Wei Lu[2] and Shangling Jui[3]

[1]University of Alberta

[2]Huawei Technologies Canada Co., Ltd.

[3]Huawei Kirin Solution, Shanghai, China

Motivation

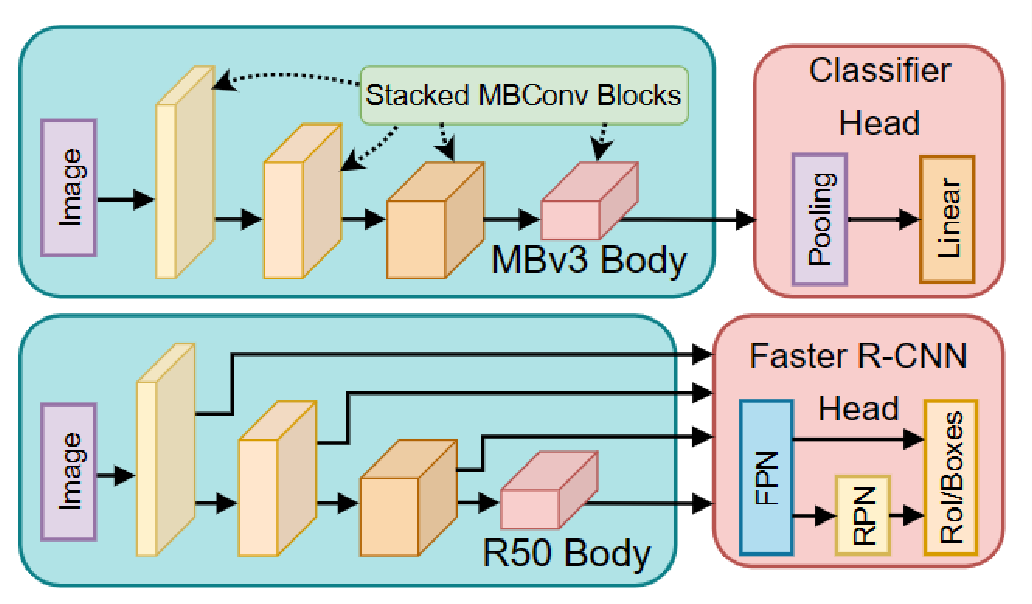
- In Neural Architecture Search (NAS), Performance Evaluation is costly.
  - Task complexity, training dataset size (#samples and resolution), etc.
- Existing methods to reduce resource bottleneck include:
  - Neural predictors, Supernets, Zero-Cost Proxies (transferable), etc.
- **Issues: currently, NAS mainly targets image classification performance**
  - Predictors mainly target common NAS benchmark tasks, e.g., NAS-Bench-101
  - Trained by CIFAR-10/100, ImageNet classification accuracies
  - Cost of enabling NAS performance evaluation for a new task is too high
- **However, in practice CV tasks/datasets are specific and diverse**
  - Segmentation, Detection, Human Pose, Super Resolution...
  - Task networks have more complex network topological features
  - Use different datasets: MS-COCO [Lin et al. 2014], MPII [Andriluka et al. 2014].

Contributions

We propose AIO-P, or **A**ll-**I**n-**O**ne **P**redictors, for multi-task NAS evaluation

- Pretrain Predictor on NAS Benchmarks (classification) but generalize NAS evaluation to specific CV tasks used in reality
- Apply ***K*-Adapters** [Wang et al. 2021] to inject domain-specific knowledge into the pretraining.
- Propose a **pseudo-labeling** scheme to generate *K*-Adapter training samples.
- Incorporate **scaling techniques** and FLOPs to augment predictor labels.
  - Transfer prediction to task-specific metric beyond accuracy or mAP.
- **Verify** AIO-P on predicting NN performance on many CV tasks
  - Pose Estimation, Object Detection, Instance/Semantic/Panoptic Segmentation.
  - Demonstrate **transferability** to different network families
  - **Applying to NAS**, optimized a proprietary Facial Recognition (FR) model.
- Open-source code and data: https://github.com/Ascend-Research/AIO-P

Task-Aware Network Representation

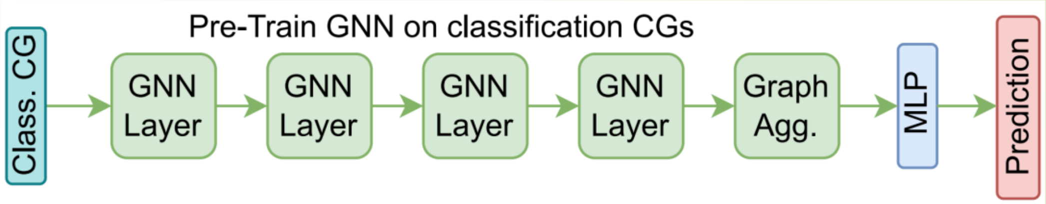


- We can use the same CV backbone network for different tasks:
  - Network body is a feature extractor.
    - Like MobileNets and ResNets.
  - Network head is task-specific.
    - Pooling + Linear for Classification.
    - Deconvolution for Pose Estimation.
    - R-CNN for Object Detection.
- Represent neural network using a Computational Graph (CG).
  - Defined by a forward-pass in TensorFlow (based on .pb file)
  - Cross-family representatability

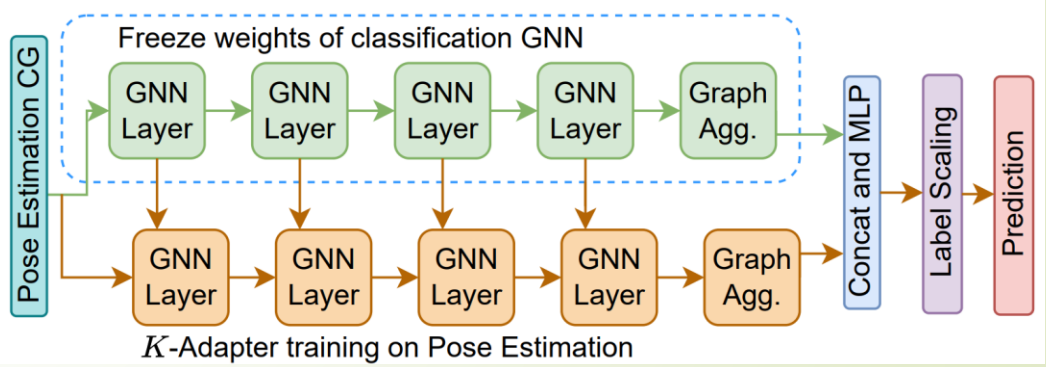
HUAWEI HISILICON

Predictor Pretraining + Knowledge Infusion

- Pretrain a simple GNN predictor.
  - Pre-trained on classification NAS benchmarks (e.g., NAS-Bench-101)
  - This is the "predictor backbone".

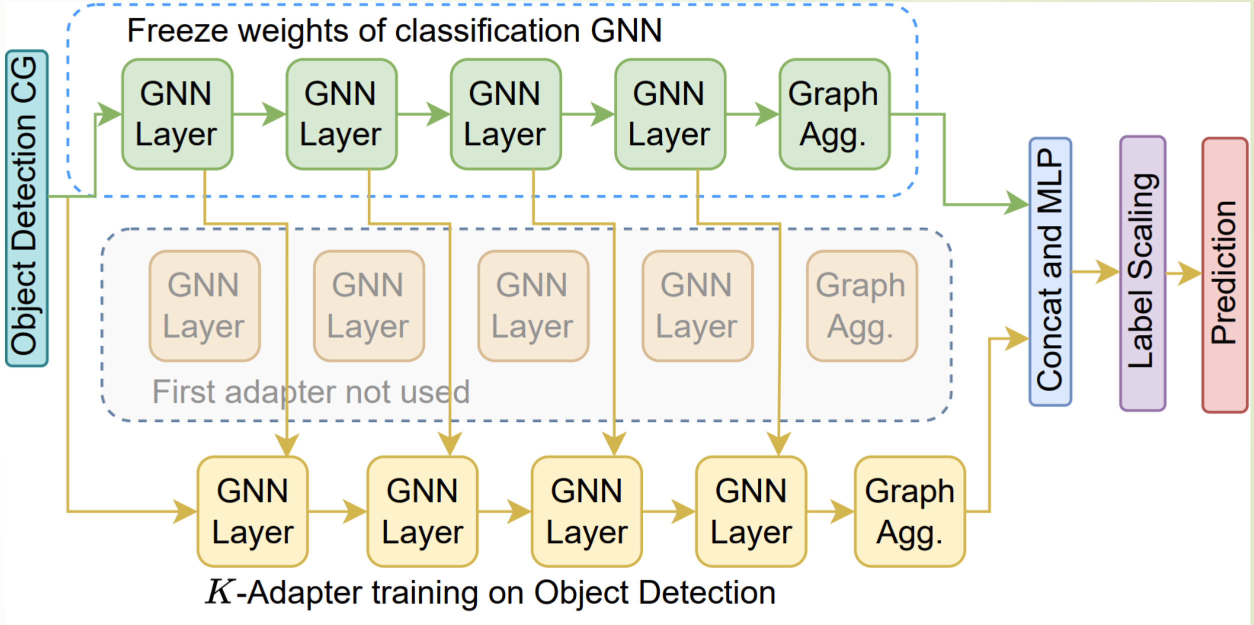


- Append a *K*-Adapter to the existing "predictor backbone"
  - Train K-Adapter on a set of CGs labeled for a new task's performance
  - Freezing weights of backbone
  - Incorporate label scaling



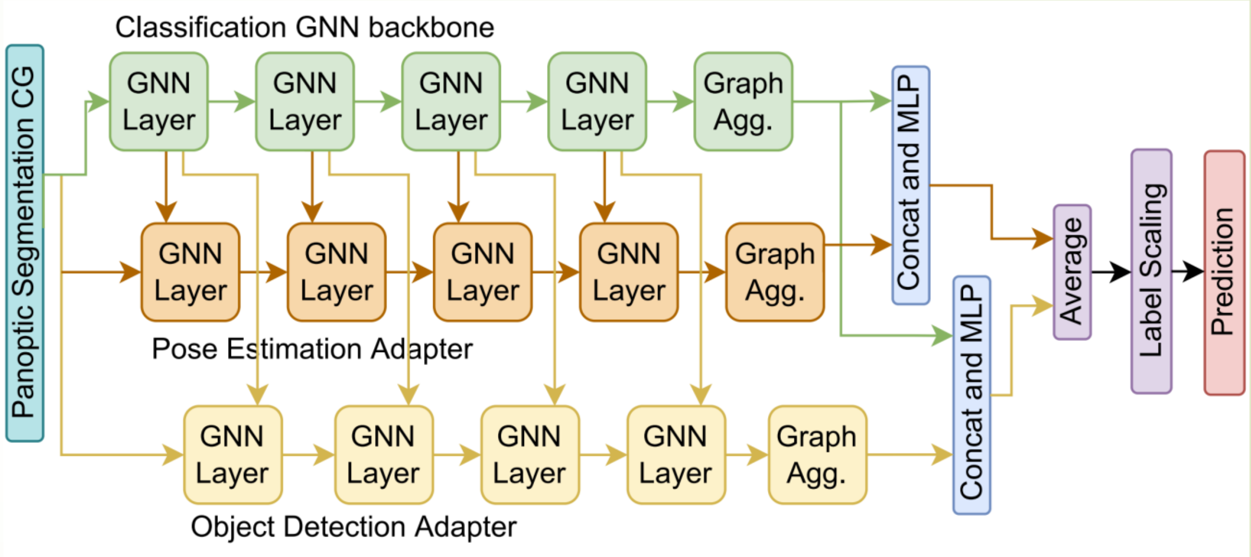
K-Adapters for Knowledge Infusion

- We can add multiple *K*-Adapters to the same predictor backbone.
  - One for each new task.
  - Trained independently with separate final MLP layers.
- Can inject knowledge from desired tasks/network families into the predictor.



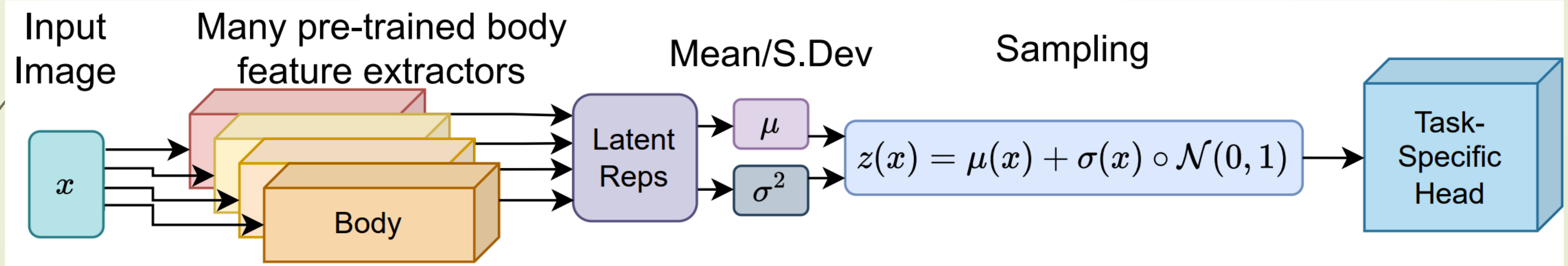
Applying to Downstream Tasks

- Downstream prediction combines all *K*-Adapters.
  - Average their predictions.
  - Apply label-scaling.
  - Fine-tune on a small number of task-specific network samples.



Pseudo-Labeling: to obtain K-Adaptor Pretraining Samples

- We need CG samples labeled on a task to pretrain *K*-Adapters.
  - Fully evaluating an individual network on a task can take hours.
- We train a shared task head that generalizes to the entire design space.

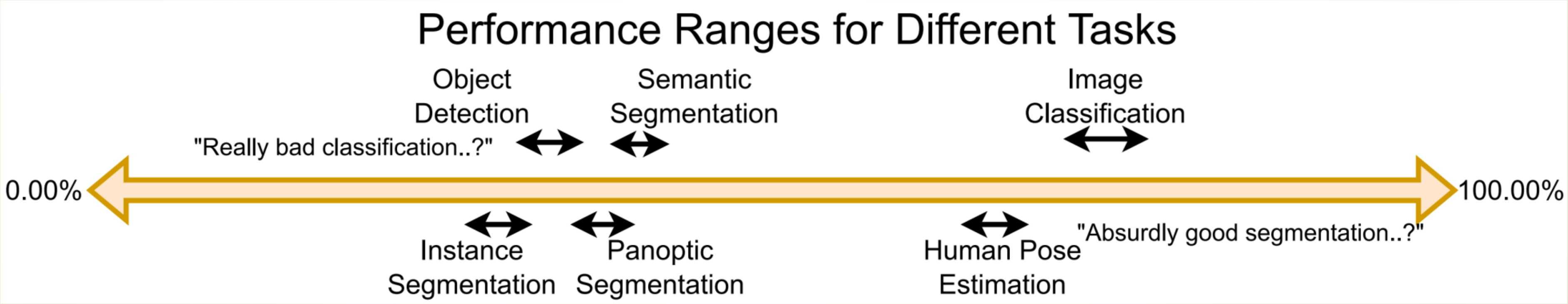


e.g. pretrained OFA networks

- Pair the shared head with an individual body to pseudo-label the network's performance on this task
  - Fine-tune body + shared head on the task dataset for a few minutes

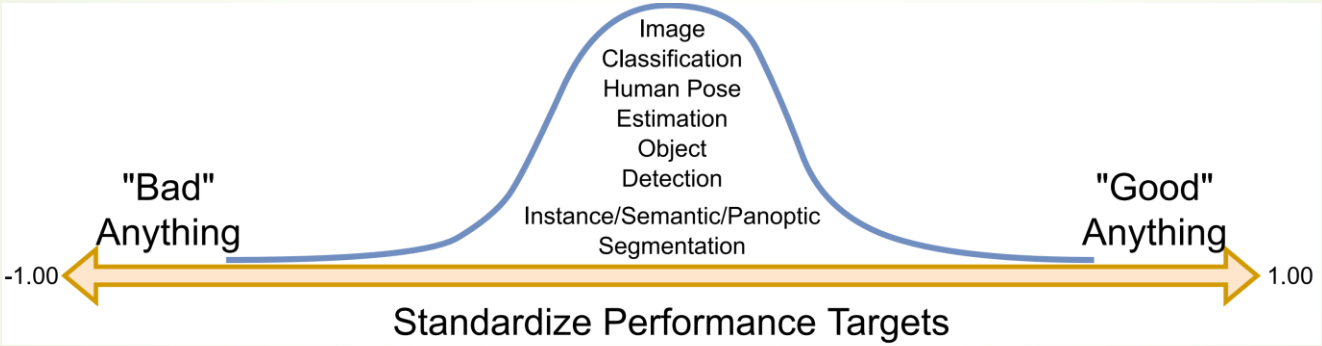
Label Scaling

- Different CV tasks have different performance metrics
  - Classification accuracy, Percentage of Correct Keypoints (PCK), mean Average Precision (mAP), mean Intersection over Union (mIoU), Panoptic Quality (PQ), etc.
- The distributions and value ranges of these metrics may vary
- How to overcome this when using *K-Adapters* for knowledge infusion?



Label Scaling

- Simple solution: Do not predict absolute values of performance metrics.
  - Use standardization to generate a unitless measure of performance.
    - Calculate mean/variance using 20 held-out samples.



- Furthermore, normalize original labels by FLOPs using the analytical equation:

$$y_F = y \cdot (\mathrm{Log}_{10}(F+1)+1)^{-1},$$

  - FLOPs are a measure of model and dataset size.
  - Has a positive correlation with performance.
  - Easy to compute.

Experimental Setup

1. **AIO-P Task Predictor Performance:**
   - Train AIO-P with 2 *K*-Adapters, then apply to a wide range of downstream tasks.
   - Measure ranking correlation (SRCC) and Mean Absolute Error (MAE).
   - Evaluate under zero-shot transfer and small fine-tuning (20 samples) contexts.
2. **Comparison with other generalizable NAS performance evaluation methods:**
   - Zero-Cost Proxies [Abdelfattah et al. 2021]
3. **Generalization to Different Unseen Model Zoos:**
   - EfficientNets, Inception Nets, DeepLab Sematic Segmentation [Chen et al. 2017], etc.
4. **Application to NAS:**
   - Successfully improved a proprietary Facial Recognition (FR) model.

Verify AIO-P's Ability on OFA NAS Benchmarks

- **Ground-truth test networks: OFA-ProxylessNAS/MBv3/ResNet50 networks + a task head fully trained on a downstream task dataset**
- **Baseline: backbone GNN pretrained on NB101 networks**, (Eq. 4/5: standardization/FLOPs transform for label scaling).
- **AIO-P: GNN on NB101 + K-Adapters** on **Pose Estimation** and **Object Detection** (**pseudo-labeling via OFA bodies+shared head**)

Spearman Ranking Correlation (SRCC)

Zero Shot Transfer

	ProxylessNAS		
Task	GNN	+Eqs. 4 & 5	AIO-P
LSP	0.593 ± 0.02	0.561 ± 0.05	**0.698** ± 0.01
MPII	0.711 ± 0.01	**0.767** ± 0.02	0.753 ± 0.01
OD	0.558 ± 0.06	0.471 ± 0.11	**0.781** ± 0.03
IS	0.599 ± 0.07	0.211 ± 0.10	**0.831** ± 0.02
SS	0.487 ± 0.03	0.262 ± 0.18	**0.735** ± 0.02
PS	0.562 ± 0.00	0.119 ± 0.12	**0.732** ± 0.03

Fine Tune (with 20 samples)

	ProxylessNAS		
Task	GNN	+Eqs. 4 & 5	AIO-P
LSP	0.610 ± 0.02	0.583± 0.07	**0.668**± 0.03
MPII	0.770 ± 0.02	**0.803**± 0.02	0.773± 0.02
OD	0.304 ± 0.46	0.589± 0.06	**0.800**± 0.05
IS	0.277 ± 0.70	0.330± 0.14	**0.894**± 0.03
SS	0.195 ± 0.33	0.562± 0.11	**0.849**± 0.03
PS	0.741 ± 0.04	0.297± 0.08	**0.868**± 0.04

Mean Absolute Error (MAE)

Zero Shot Transfer

	ProxylessNAS		
Task	GNN	+Eqs. 4 & 5	AIO-P
LSP	27.27 ± 0.39%	0.72 ± 0.14%	**0.70** ± 0.23%
MPII	8.10 ± 0.38%	**0.34** ± 0.07%	0.42 ± 0.13%
OD	59.53 ± 0.41%	1.15 ± 0.46%	**0.63** ± 0.09%
IS	62.00 ± 0.34%	0.93 ± 0.18%	**0.52** ± 0.14%
SS	53.07 ± 0.37%	0.71 ± 0.22%	**0.50** ± 0.06%
PS	56.19 ± 0.35%	0.76 ± 0.07%	**0.50** ± 0.10%

Fine Tune (with 20 samples)

	ProxylessNAS		
Task	GNN	+Eqs. 4 & 5	AIO-P
LSP	0.55 ± 0.39%	0.56 ± 0.04%	**0.48**± 0.02%
MPII	0.43 ± 0.22%	0.28 ± 0.02%	**0.26**± 0.02%
OD	0.90 ± 0.16%	0.74 ± 0.07%	**0.53**± 0.04%
IS	0.72 ± 0.15%	0.75 ± 0.09%	**0.33**± 0.03%
SS	0.68 ± 0.12%	0.58 ± 0.04%	**0.33**± 0.03%
PS	0.53 ± 1.00%	0.62 ± 0.04%	**0.33**± 0.04%

Comparing to Other Transferable Performance Evaluation Methods in NAS

- Consider several Zero-Cost Proxies and FLOPs.
- Measure SRCC and compare.
  - ZCP performance is inconsistent per search space, sometimes negative.
- AIO-P achieves positive SRCC above 0.65 for all three search spaces even under zero-shot transfer
  - Can further enhance performance with small-sample fine-tuning.

Space	Synflow	Jacov	Fisher	Gradient Norm	Snip	FLOPs	AIO-P	AIO-P FT
PN-SS	0.022 $\pm$ 0.07	-0.023 $\pm$ 0.13	0.050 $\pm$ 0.07	0.141 $\pm$ 0.06	-0.082 $\pm$ 0.07	0.608 $\pm$ 0.01	*0.735* $\pm$ 0.02	**0.849** $\pm$ 0.03
MBv3-SS	-0.309 $\pm$ 0.07	0.042 $\pm$ 0.08	0.022 $\pm$ 0.06	0.040 $\pm$ 0.06	0.188 $\pm$ 0.04	0.445 $\pm$ 0.02	*0.689* $\pm$ 0.02	**0.822** $\pm$ 0.03
R50-SS	-0.255 $\pm$ 0.09	0.141 $\pm$ 0.10	0.126 $\pm$ 0.059	0.354 $\pm$ 0.08	0.036 $\pm$ 0.07	*0.661* $\pm$ 0.02	0.660 $\pm$ 0.02	**0.677** $\pm$ 0.03

Paper Tab. 5: Comparison to Zero-Cost Proxies

Transferability to Different Types of Classical Model Zoos

- What is a 'model zoo'?
  - Handful of task networks not part of a NAS Benchmark or search space.
    - E.g., **EfficientNet-{B0-B7} models**.
    - E.g., **Inception-{v1-v4}**
- Predict performance of these out-of-distribution networks.
- AIO-P achieves SRCC > 0.9 on DeepLab Semantic Segmentation.
  - Leverage Eq. 5, FLOPs transform.
- Perfect SRCC=1.0 on EfficientNets.

Model Zoo	#Archs	AIO-P w/o Eq. 5	AIO-P
DeepLab-ADE20k	5	0.127 ± 0.255	$\mathbf{0.991} \pm 0.016$
DeepLab-Pascal	6	0.392 ± 0.088	$\mathbf{0.939} \pm 0.035$
DeepLab-Cityscapes	8	0.572 ± 0.031	$\mathbf{0.925} \pm 0.024$
Slim-ResNets	6	-0.577 ± 0.183	$\mathbf{0.920} \pm 0.106$
Slim-Inception	5	-0.700 ± 0.316	$\mathbf{0.980} \pm 0.040$
Slim-MobileNets	5	-0.500 ± 0.000	$\mathbf{0.400} \pm 0.535$
Slim-EfficientNets	8	$\mathbf{1.000} \pm 0.000$	$\mathbf{1.000} \pm 0.000$

Paper Tab. 12: SRCC of AIO-P on Model Zoos.

Applying AIO-P to NAS: a reality check

	Full	Simple	Lighted	Dark	FLOPs
Base Model Pr	**96.3%**	**98.7%**	**97.9%**	96.5%	563M
AIO-P Search Pr	96.1%	**98.7%**	**97.9%**	**96.7%**	486M
Base Model Rc	**91.9%**	**98.3%**	**96.8%**	92.6%	563M
AIO-P Search Rc	91.1%	98.2%	96.6%	**93.2%**	486M

Paper Tab. 13: Optimizing FR to preserve

Precision (Pr) and Recall (Rc) while reducing FLOPs.

- **Grand goal of pretraining AIO-P**
  - **Fast and low-cost NAS evaluation on any network type and for any task**
- Pair AIO-P with a search algorithm.
  - Optimized a proprietary mobile Facial Recognition (FR) network.
  - Aim to preserve performance while making the model light-weight.
- Reduced FLOPs by over 13% while still maintaining precision and recall.

Conclusion

Propose AIO-P, or **A**ll-**I**n-**O**ne **P**redictors for transferrable task performance prediction in NAS.

- Inject knowledge from different tasks into a GNN predictor using ***K*-Adapters**.
- Develop a **pseudo-labeling** scheme to generate *K*-Adapter training data.
- Incorporate **label scaling** to learn a unitless measure of performance.
  - For dealing with diverse tasks with different metric ranges, e.g., mAP vs. PCK.
- **Evaluate** the performance of AIO-P in several contexts:
  - Task-transferability tests measuring SRCC and MAE.
  - Compared to Zero-Cost Proxies.
  - Classification and Semantic Segmentation Model Zoos.
  - Application to NAS: Optimizing proprietary mobile networks.
- **Open-source** our code and data to advance the field.

References

- Lin et al., "Microsoft COCO: Common Objects in Context." In ECCV 2014.
- Andriluka et al., "2D Human Pose Estimation: New Benchmark and State of the Art Analysis.", CVPR 2014.
- Wang et al., "K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters." In ACL-IJCNLP 2021.
- Abdelfattah et al., "Zero-Cost Proxies for Lightweight NAS." In ICLR 2021.
- Chen et al., "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution and Fully Connected CRFs." In IEEE TPAMI.

UNIVERSITY OF ALBERTA QUAECUMQUE VERA

HUAWEI HISILICON