ETH zürich

THE UNIVERSITY OF SYDNEY

KU LEUVEN

ECCV'20 ONLINE 23-28 AUGUST 2020

DHP: Differentiable Meta Pruning via HyperNetworks

Yawei Li 

Shuhang Gu 

Kai Zhang 

Luc Van Gool 

Radu Timofte 

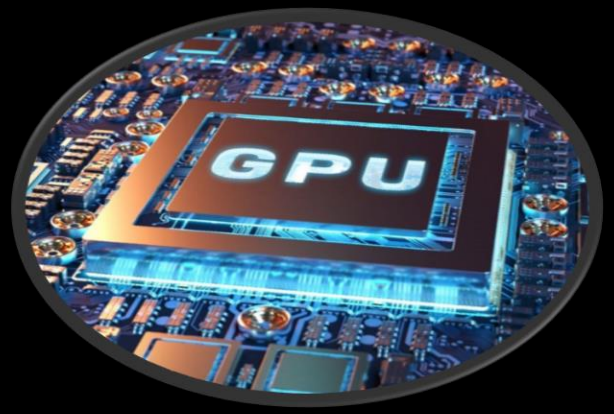
Introduction

DHP: Differentiable Meta Pruning via HyperNetworks

Computer Vision

Speech Recognition

Natural Language Processing

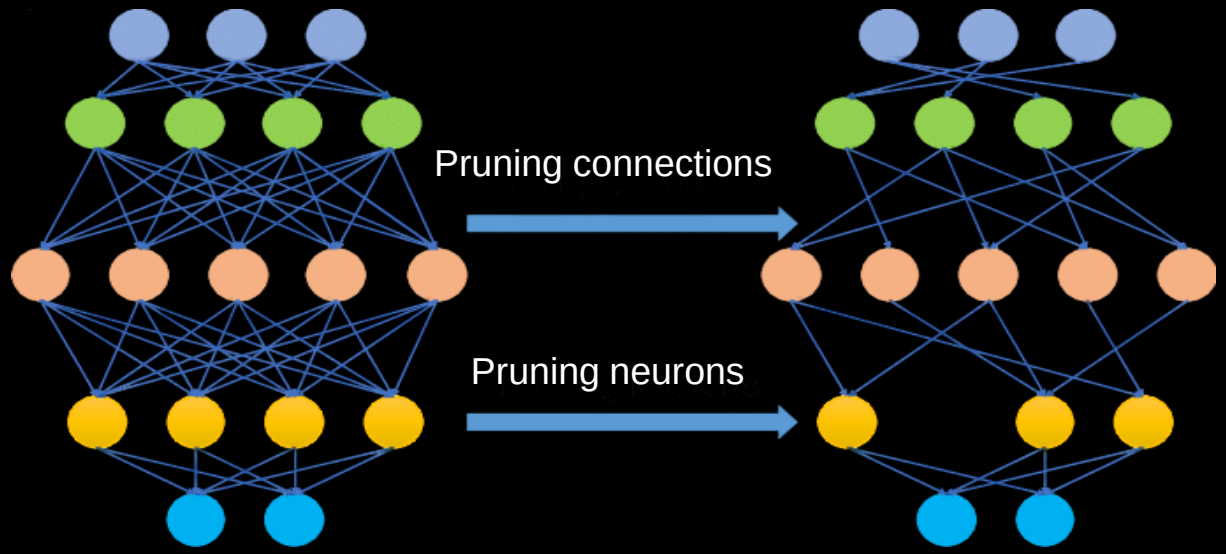


Computation

Storage

Transmission

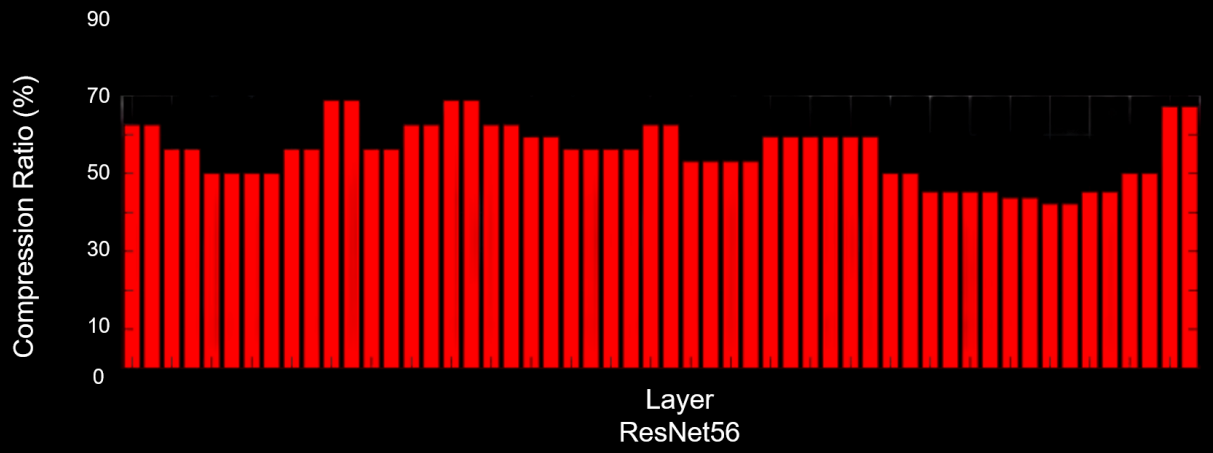
DHP: Differentiable Meta Pruning via HyperNetworks



Network Pruning

Unstructured pruning: pruning the connections

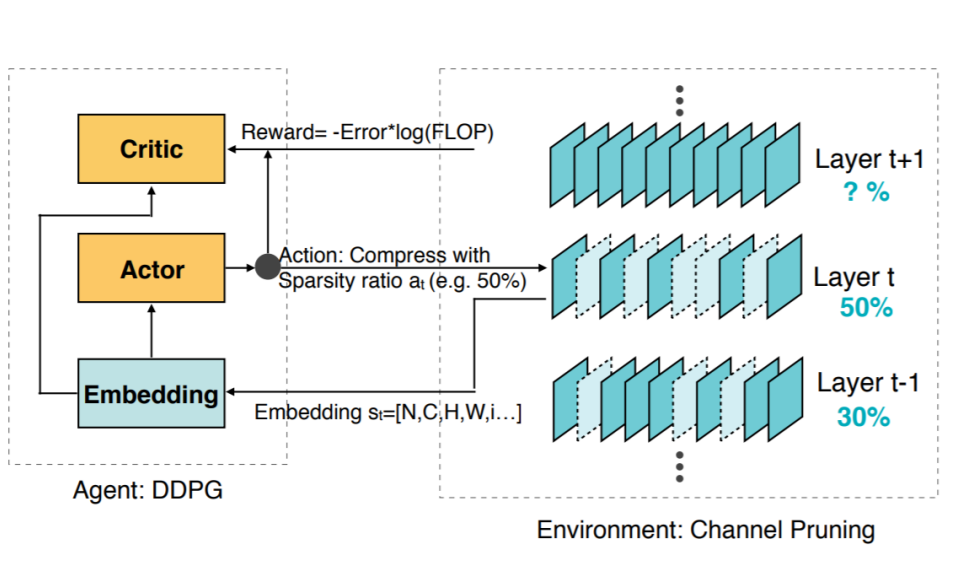
Structured pruning: pruning the neurons



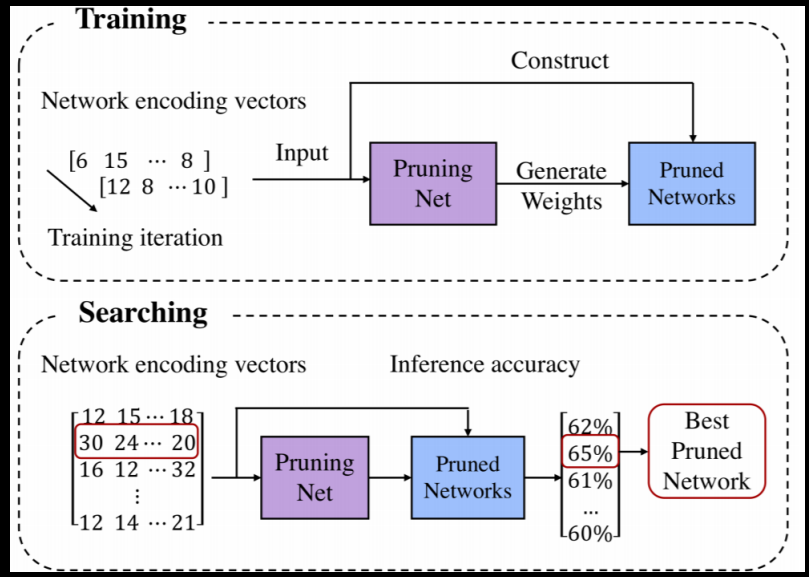
Automatic Network Pruning: determine the pruning ratio for each layer automatically

DHP: Differentiable Meta Pruning via HyperNetworks

AMC
Reinforcemet Learning



MetaPruning
Evolutionary Algorithm



Problem:

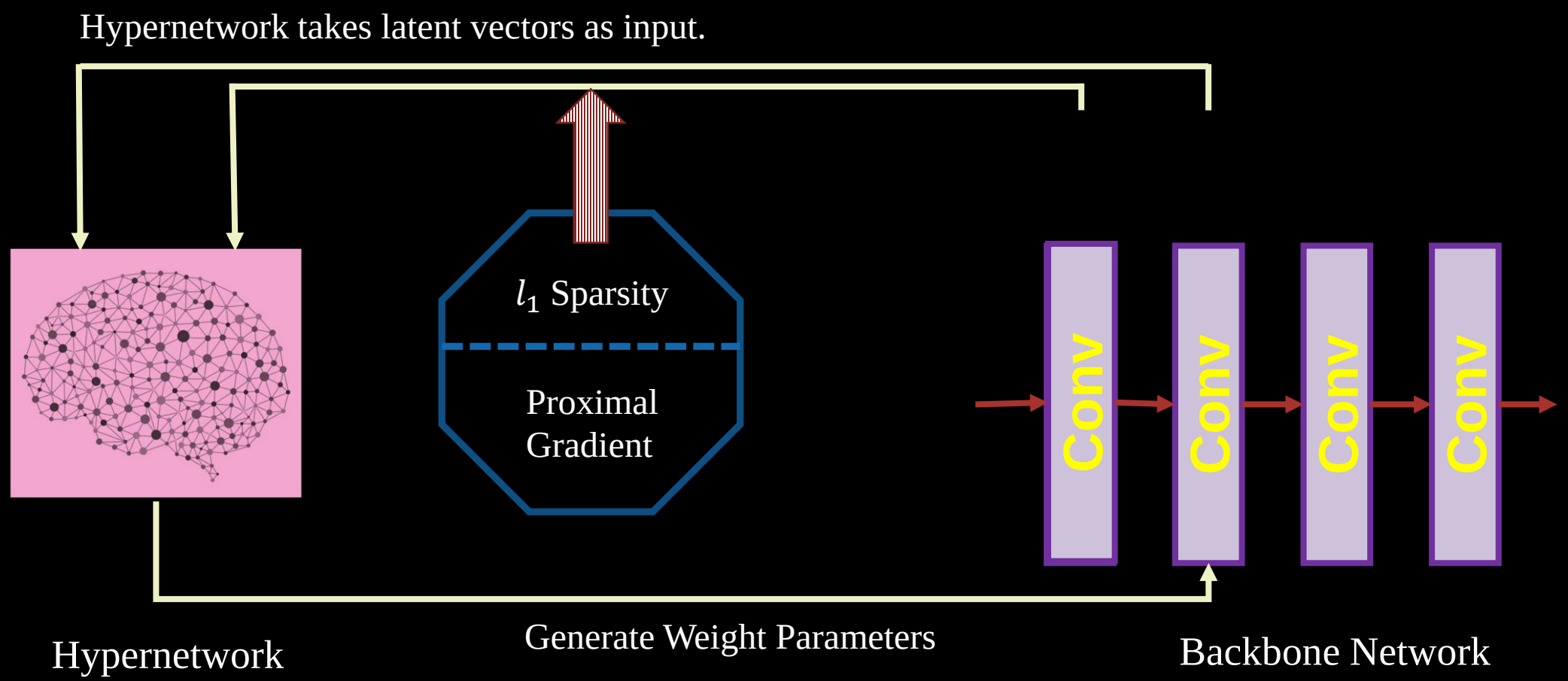
1. None differentiability
2. Convergence

Contribution:

1. A new architecture of hypernetwork is designed.
2. A differentiable automatic networking pruning method is proposed.
3. The potential of automatic network pruning as fine-grained architecture search is revealed.

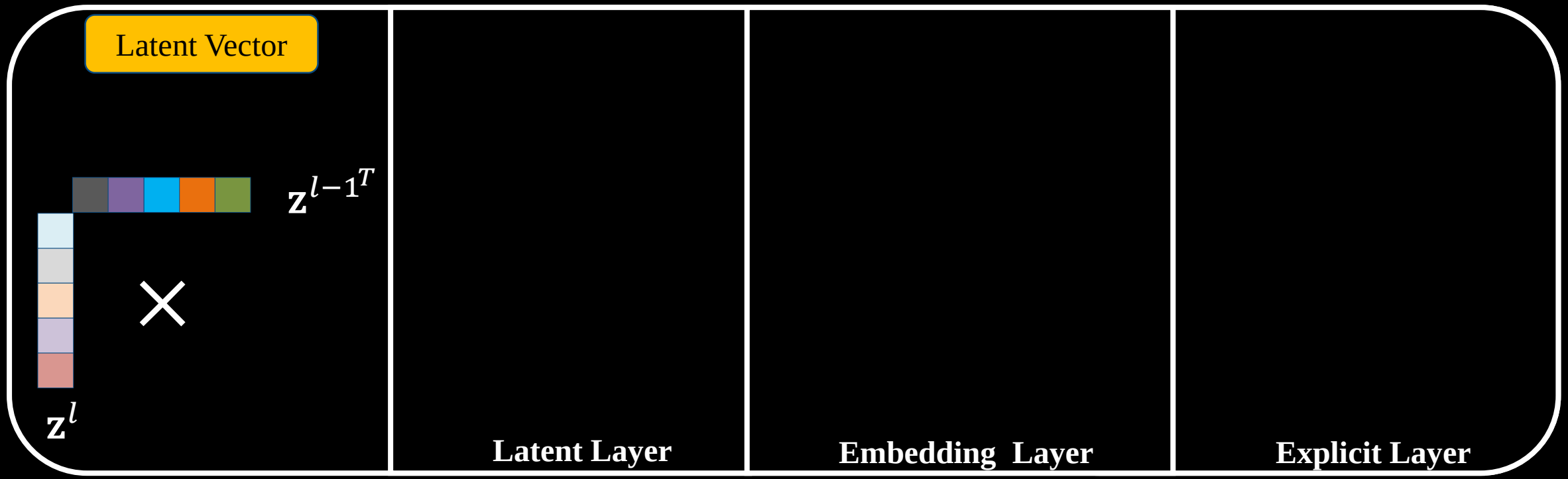
Method

DHP: Differentiable Meta Pruning via HyperNetworks



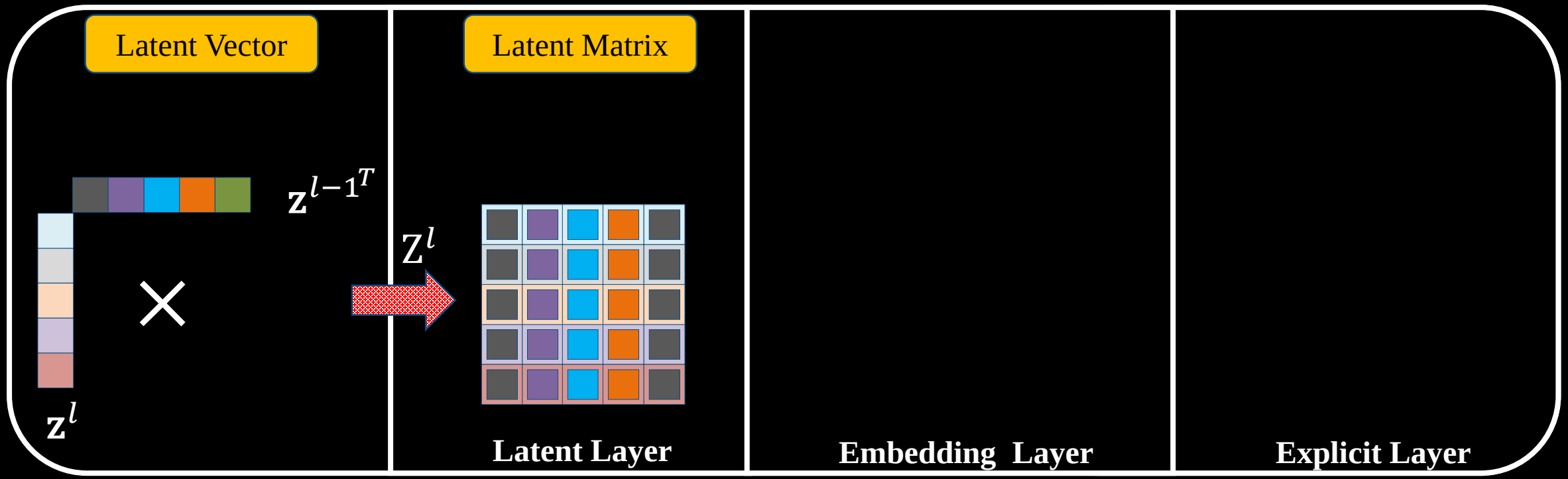
The Overview of the Proposed DHP Method

DHP: Differentiable Meta Pruning via HyperNetworks



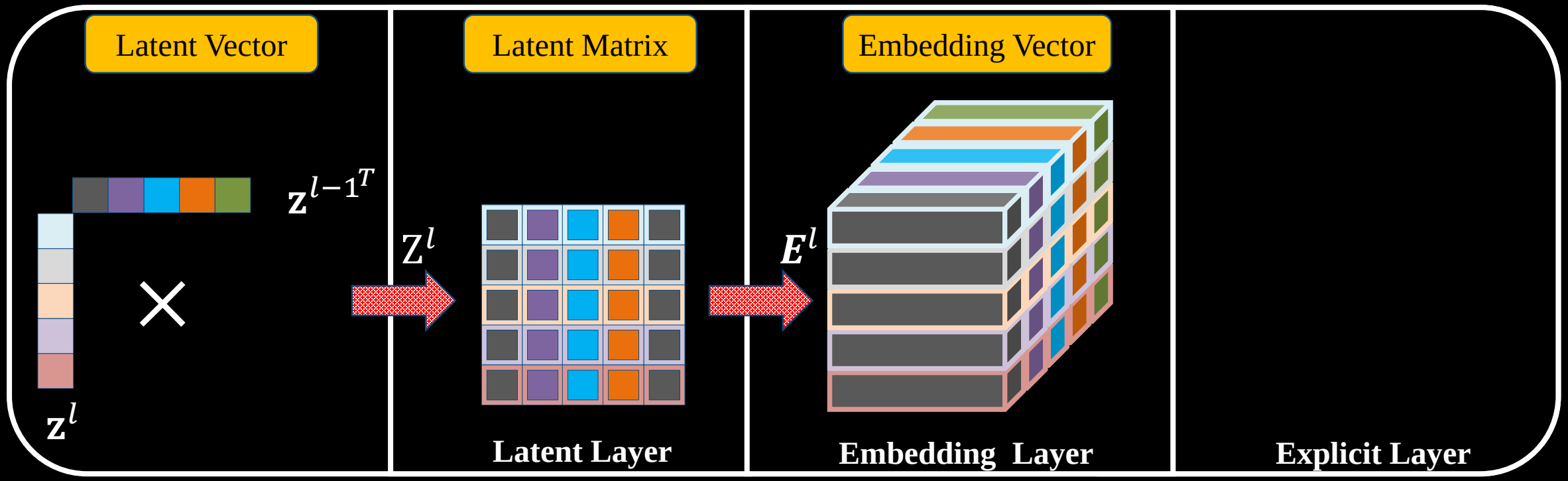
The Specifically Designed Hypernetwork.

DHP: Differentiable Meta Pruning via HyperNetworks



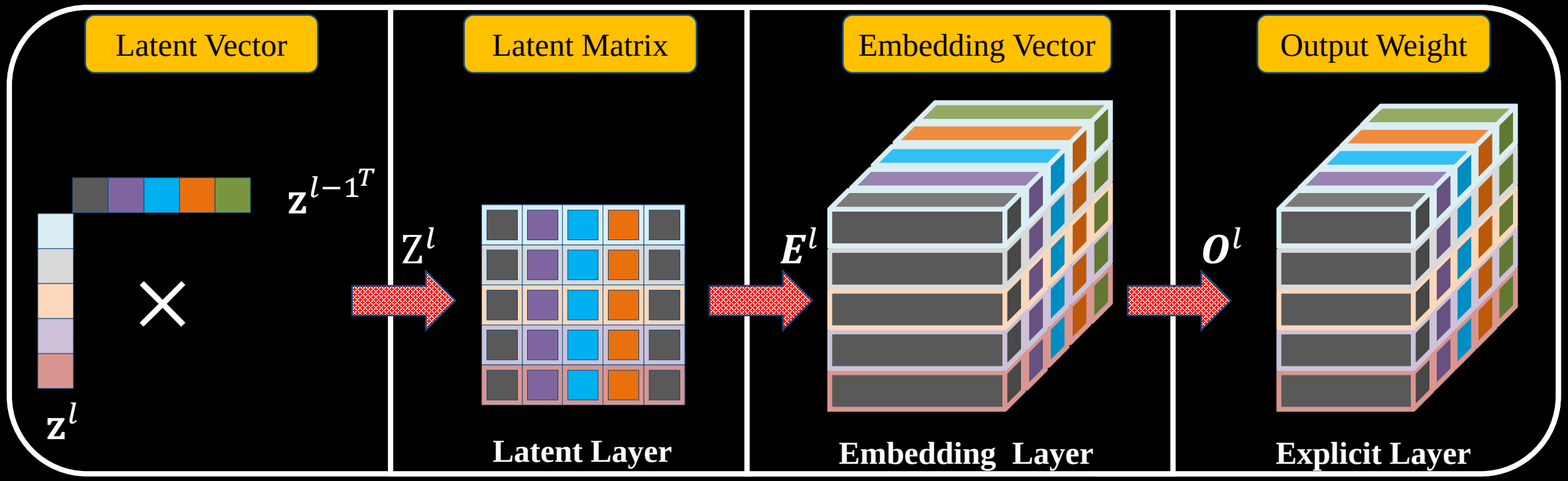
The Specifically Designed Hypernetwork.

DHP: Differentiable Meta Pruning via HyperNetworks



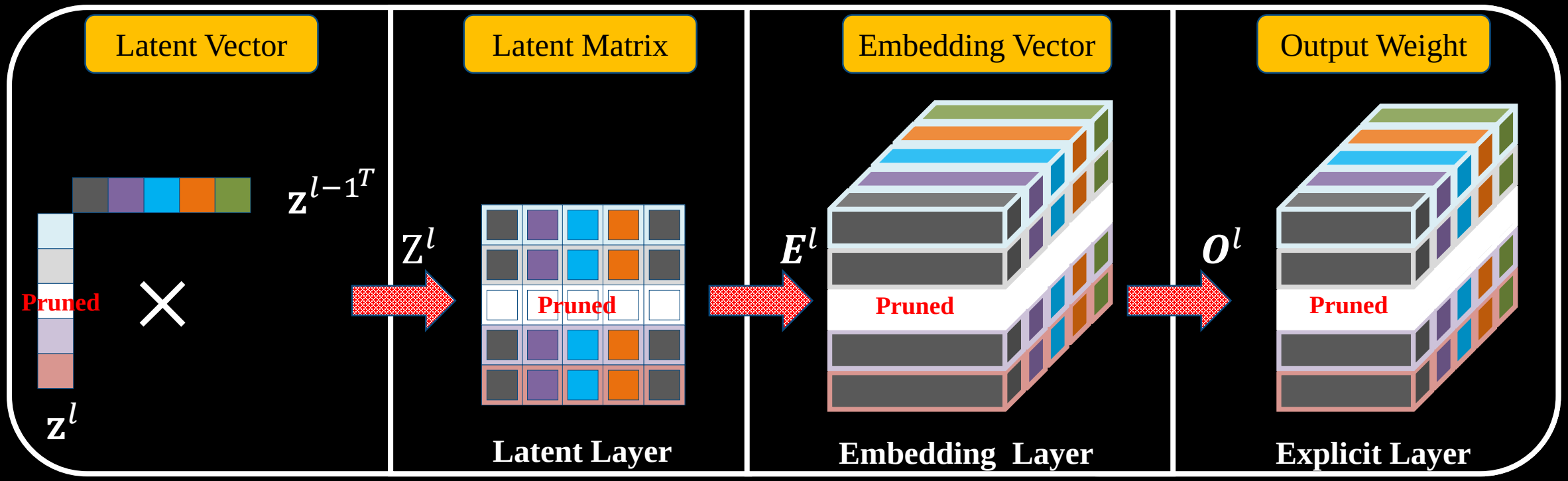
The Specifically Designed Hypernetwork.

DHP: Differentiable Meta Pruning via HyperNetworks



The Specifically Designed Hypernetwork.

DHP: Differentiable Meta Pruning via HyperNetworks



The Specifically Designed Hypernetwork.

DHP: Differentiable Meta Pruning via HyperNetworks

Loss Function

Original Loss

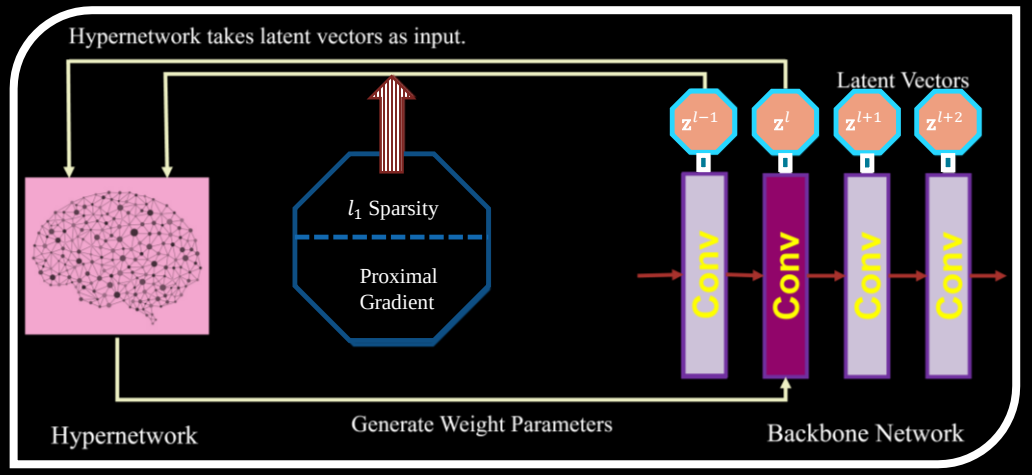
Sparsity Regularization

$$\min_{\mathbf{W},\mathbf{B},\mathbf{z}} \mathcal{L}\Big(\mathbf{y}, f\big(\mathbf{x}; h(\mathbf{z}; \mathbf{W}, \mathbf{B})\big)\Big) + \gamma\mathcal{D}(\mathbf{W}) + \gamma\mathcal{D}(\mathbf{B}) + \lambda\mathcal{R}(\mathbf{z})$$

Weight Decay

Sparsity

$$\mathcal{R}(\mathbf{z}) = \sum_{l=1}^{L} \|\mathbf{z}^l\|_1$$



Gradient descent

Proximal Operator

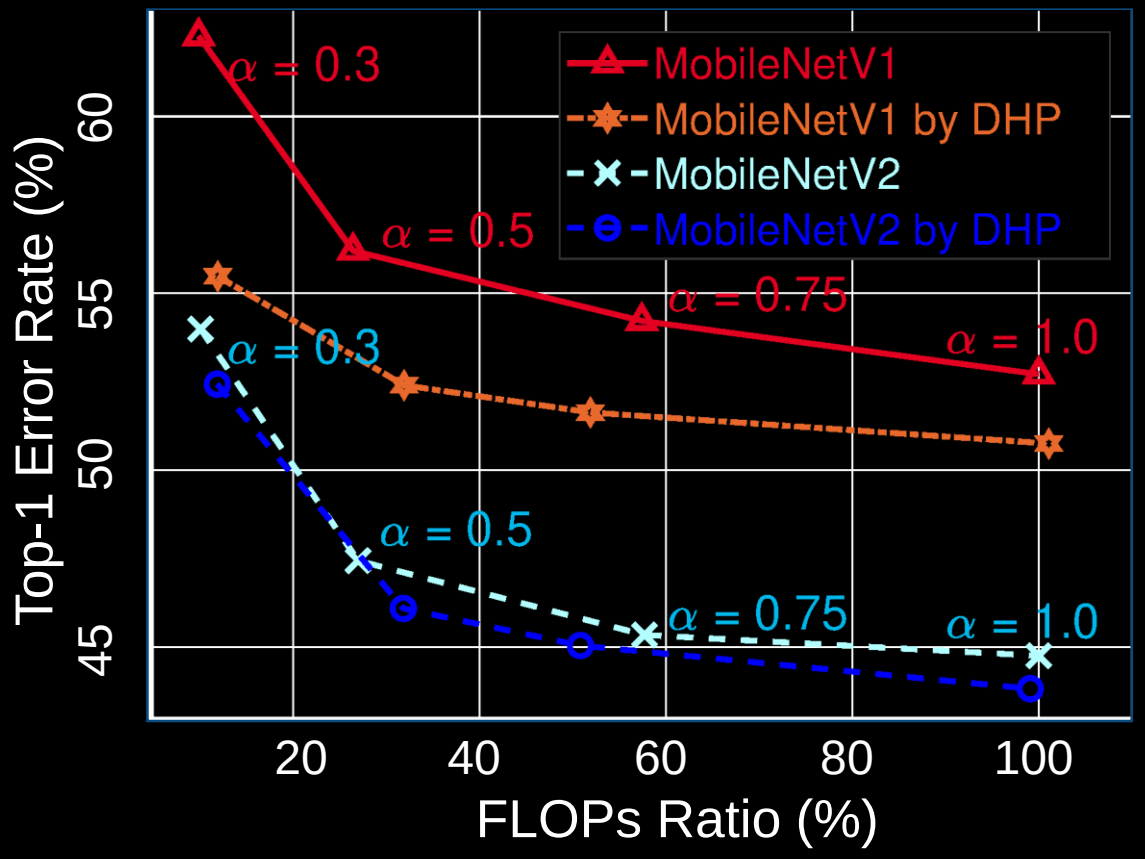
$$\mathbf{z}[k+1] = \mathbf{prox}_{\lambda\mu\mathcal{R}}\Big(\mathbf{z}[k] - \lambda\mu\nabla\mathcal{L}\big(\mathbf{z}[k]\big)\Big)$$

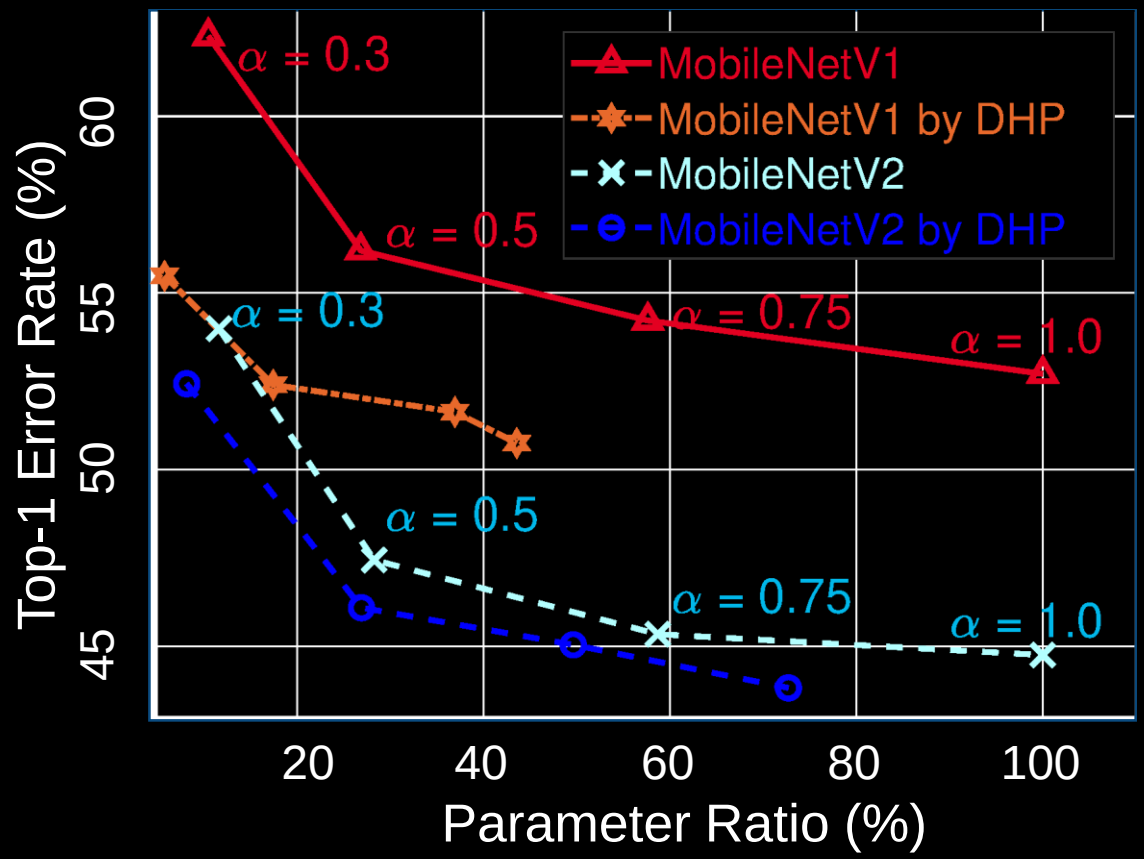
Proximal step

The Proximal Gradient Algorithm

RESULTS

DHP: Differentiable Meta Pruning via HyperNetworks





Observation: The original model with different width multipliers α is set as the baseline. The DHP models outperforms the original at all the operating points

DHP: Differentiable Meta Pruning via HyperNetworks

Table 1. Results on CIFAR10 image classification. DHP outperforms the compared methods under comparable model complexity.

Network Top-1 Error (%)	Compression Method	Top-1 Error (%)	FLOPs Ratio (%)	Parameter Ratio (%)
ResNet-56 7.05	Variational [63]	7.74	79.70	79.51
	Pruned-B [30]	6.94	72.40	86.30
	NISP [60]	6.99	56.39	57.40
	DHP-50 (Ours)	**6.42**	**50.96**	**58.42**
	CaP [43]	6.78	50.20	--
	ENC [25]	7.00	50.00	--
	AMC [19]	8.10	50.00	--
	KSE [34]	6.77	48.00	45.27
	FPGM [18]	6.74	47.70	--
	GAL-0.8 [36]	8.42	39.80	34.10
	DHP-38 (Ours)	**7.06**	**39.07**	**41.10**
ResNet-164 4.97	Hinge [32]	5.40	53.61	70.34
	SSS [24]	5.78	53.53	84.75
	DHP-50 (Ours)	**5.22**	**51.67**	**50.97**
	Variational [63]	6.84	50.92	43.30
	DHP-20 (Ours)	**6.30**	**21.78**	**20.46**

Observation: DHP outperforms the compared methods under comparable model complexity.

DHP: Differentiable Meta Pruning via HyperNetworks

Table 2. Results on Tiny-ImageNet image classification. DHP-24-2 shoots lower error rates than the original model.

Network Top-1 Error (%)	Compression Method	Top-1 Error (%)	FLOPs Ratio (%)	Parameter Ratio (%)
MobileNetV1 52.71	**DHP-24-2 (Ours)**	50.75	101.08	43.58
	MobileNetV1-0.75	54.22	57.42	57.64
	MetaPruning [40]	54.48	56.77	88.14
	DHP-50 (Ours)	51.63	51.91	36.95
MobileNetV2 44.75	**DHP-24-2 (Ours)**	43.82	99.09	72.72
	DHP-10 (Ours)	52.43	11.92	6.50
	MetaPruning [40]	56.72	11.00	90.27
	MobileNetV2-0.3	53.99	10.09	11.64

Observation: On Tiny-ImageNet, DHP-24-2 shoots lower error rates than the original model.

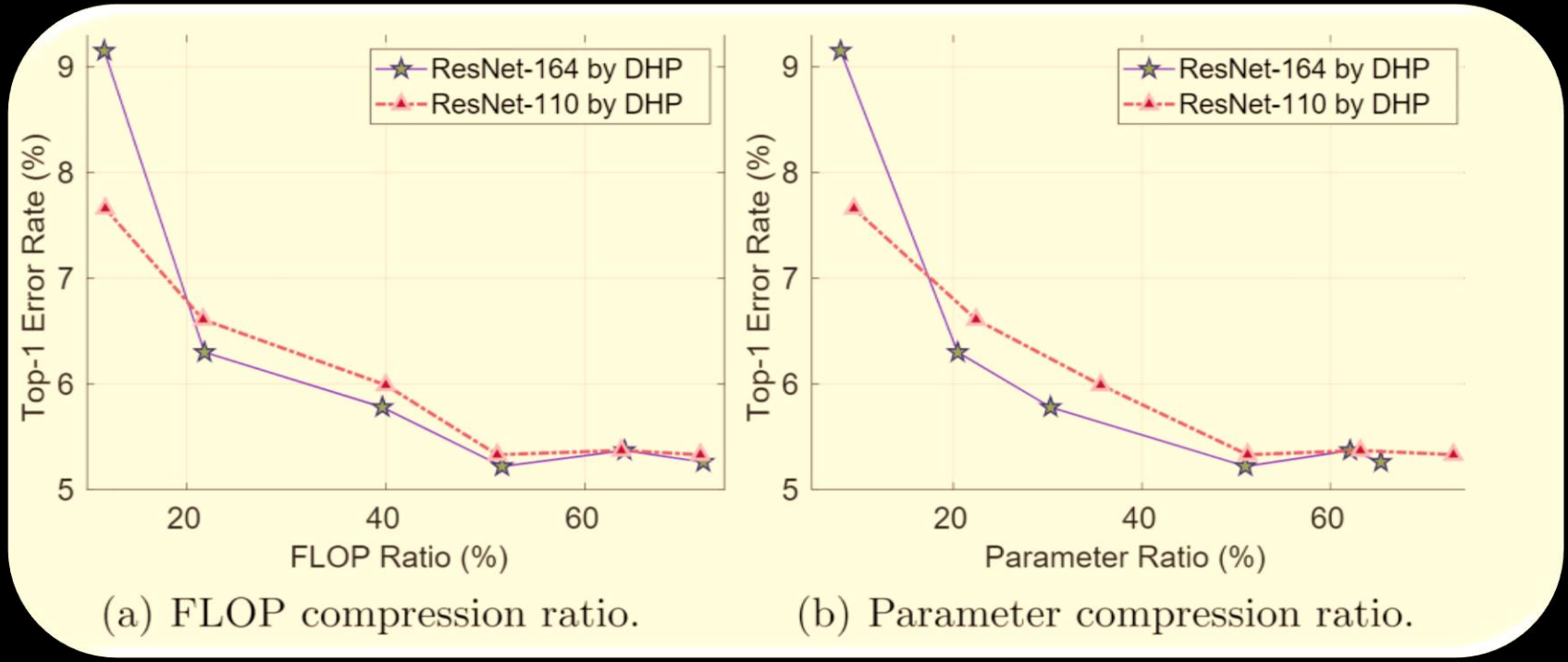
Table 2. Comparison between l_1 norm and l_2 norm regularizer. The experiments are done on ResNet.

Layer	Regularizer	Top1 Error	FLOPs	Params
20	l_1	8.46	51.80	56.13
	l_2	8.66	51.59	54.19
110	l_1	5.73	51.62	54.13
	l_2	5.77	51.37	72.37
164	l_1	5.22	51.67	50.97
	l_2	5.18	50.87	60.66

Observation: Compared with l_1 regularizer, slightly worse results can be observed for l_2 regularizer.

DHP: Differentiable Meta Pruning via HyperNetworks

Table 2. Top-1 error vs. FLOP and parameter compression ratio on ResNet-164 andResNet-110.



Observation: When the compression ratio is not too severe (above 50%), the accuracy does not drop too much.

DHP: Differentiable Meta Pruning via HyperNetworks

Table 3. Results on SRResNet for image super-resolution. The upscaling factor is ×4. Runtime is averaged for Urban100. Maximum GPU memory consumption is reported for Urban100. FLOPs is reported for a 128×128 image patch. DHP achieves significant reduction of runtime.

Method	PSNR [dB]					FLOPs [G]	Params [M]	Runtime [ms]	GPU mem [GB]
	Set5	Set14	B100	Urban100	DIV2K				
Baseline	32.03	28.50	27.52	25.88	28.85	32.83	1.54	**34.73**	0.6773
Clustering [52]	31.93	28.44	27.47	25.71	28.75	32.83	0.34	31.07	0.8123
Factor-SIC3 [56]	31.86	28.38	27.40	25.58	28.65	20.83	0.81	102.51	1.4957
DHP-60 (Ours)	31.97	28.47	27.48	25.76	28.79	20.29	0.95	**27.91**	0.5923
Basis-32-32 [33]	31.90	28.42	27.44	25.65	28.69	19.77	0.74	45.73	0.9331
Factor-SIC2 [56]	31.68	28.32	27.37	25.47	28.58	18.38	0.66	74.66	1.1201
Basis-64-14 [33]	31.84	28.38	27.39	25.54	28.63	17.49	0.60	36.75	0.6741
DHP-40 (Ours)	31.90	28.45	27.47	25.72	28.75	13.71	0.64	**22.71**	0.4907
DHP-20 (Ours)	31.77	28.34	27.40	25.55	28.60	7.77	0.36	**14.74**	0.3795

DHP: Differentiable Meta Pruning via HyperNetworks

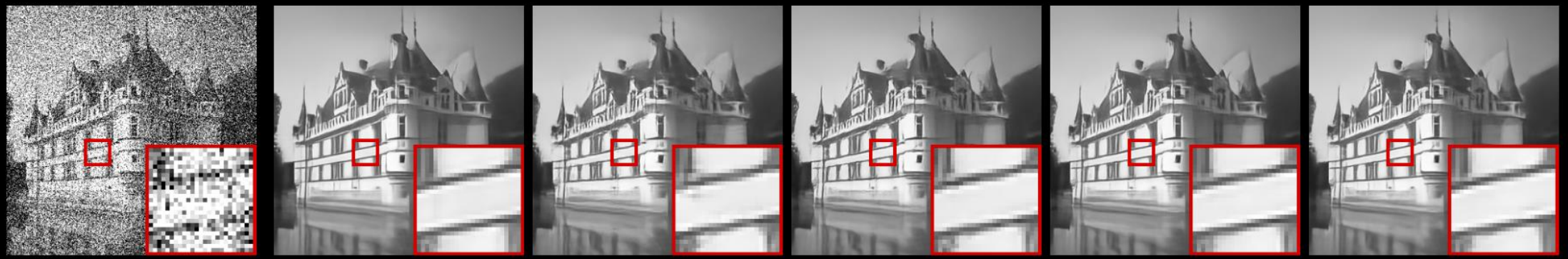
Table 4. Results on DnCNN for image denoising. The noise level is 70. Runtime and maximum GPU memory are reported for BSD68. FLOPs is reported for a 128 × 128 image. DHP achieves significant reduction of runtime.

Method	PSNR [dB]		FLOPs [G]	Params [M]	Runtime [ms]	GPU mem [GB]
	BSD68	DIV2K				
Baseline	24.93	26.73	9.13	0.56	**23.38**	0.1534
Clustering [52]	24.9	26.67	9.13	0.12	21.97	0.2973
DHP-60 (Ours)	24.91	26.69	5.65	0.34	**18.9**	0.1443
DHP-40 (Ours)	24.89	26.65	3.83	0.23	**14.62**	0.1194
Factor-SIC3 [56]	24.97	26.83	3.54	0.22	125.46	0.591
Group [47]	24.88	26.64	3.34	0.2	25.69	0.1807
Factor-SIC2 [56]	24.93	26.76	2.38	0.15	84.17	0.4149
DHP-20 (Ours)	24.84	26.58	2.01	0.12	**10.72**	0.0869

DHP: Differentiable Meta Pruning via HyperNetworks

PSNR/FLOPs/Runtime	32.85/28.59/14.10	32.50/28.59/19.75	32.65/19.82/14.71	32.24/19.28/25.49	32.64/17.61/5.40
(a) LR	(b) EDSR	(d) Cluster	(c) Basis	(f) Factor	(e) DHP

Single image super-resolution visual results. PSNR and FLOPs measured onthe image. Runtime averaged on Set5.



PSNR/FLOPs/Runtime	25.60/1.08/7.27	25.30/1.08/9.66	25.37/0.49/40.36	25.51/0.47/9.00	25.57/0.45/6.09
(a) Noisy	(b) UNet	(f) Cluster	(d) Factor	(e) Group	(c) DHP

Image denoising visual results. PSNR and FLOPs measured on the image. Runtime averaged on B100.

DHP: Differentiable Meta Pruning via HyperNetworks

Thank you for the attention!
Welcome to our Poster Session!

Code

ETH zürich