Hypercorrelation Squeeze for Few-Shot Segmentation

ICCV 2021 OCTOBER 11-17 VIRTUAL

POSTECH
POHANG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Juhong Min Dahyun Kang Minsu Cho
Pohang University of Science and Technology (POSTECH)

project page arXiv code

Problem definition and motivation

- **Few-shot segmentation:**

  Segmenting target region from a query image given a few annotated examples

- **Meta-learning:**

  Learning to learn to perform well on diverse tasks, *e.g.*, episodic training

  A dataset has training/test sets which are disjoint with respect to object classes
  Each set consist of multiple episodes, composed of *support* and *query* sets

- **Limitations of existing approaches:**

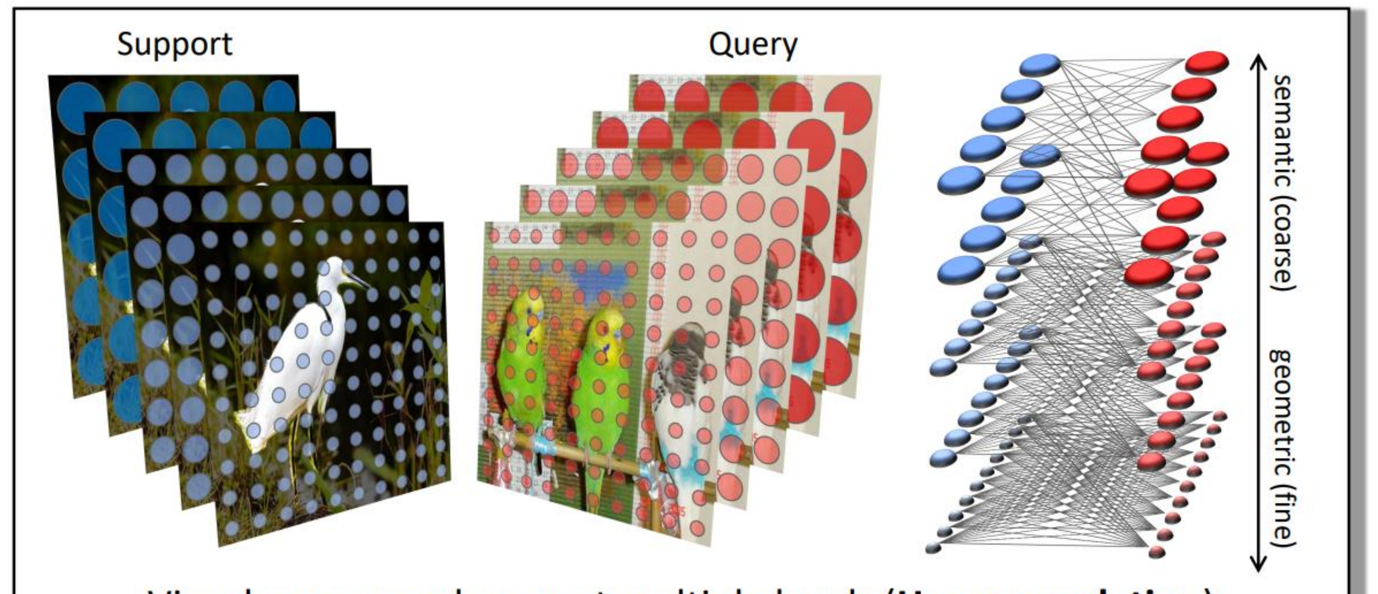
  Mostly adopt on prototype-based approach which loses spatial structure

  Hardly explore diverse levels of feature representation from a pretrained CNN

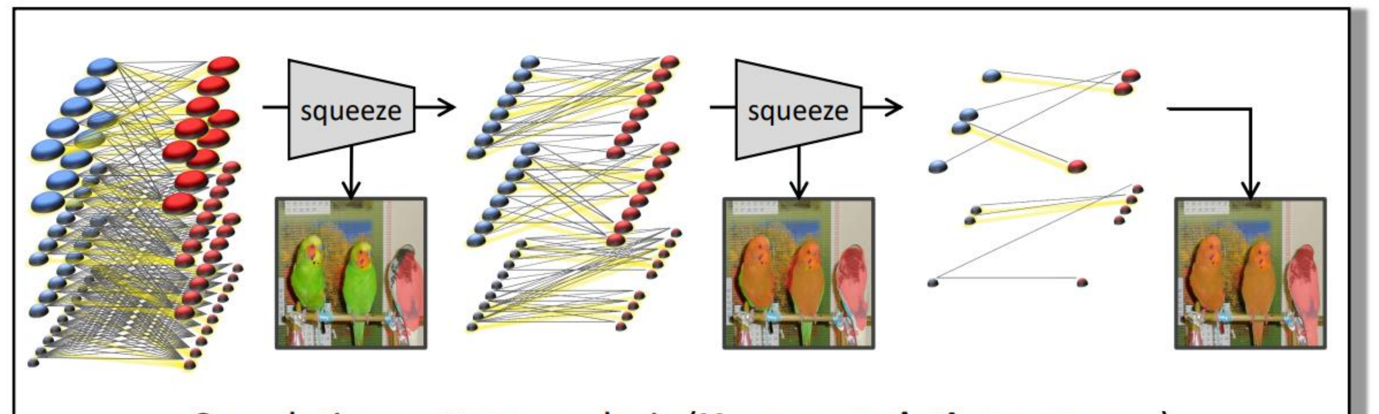
  Fail to capture relational patterns in complex pair-wise feature correlations

Contributions

1. Present the **Hypercorrelation Squeeze Networks** with deep 4D convs
2. Propose effective and efficient **center-pivot 4D conv** kernels
3. Achieve **SOTA** on three standard benchmarks of few-shot segmentation



Visual correspondences at multiple levels (**Hypercorrelation**)



Correlation pattern analysis (**Hypercorrelation squeeze**)

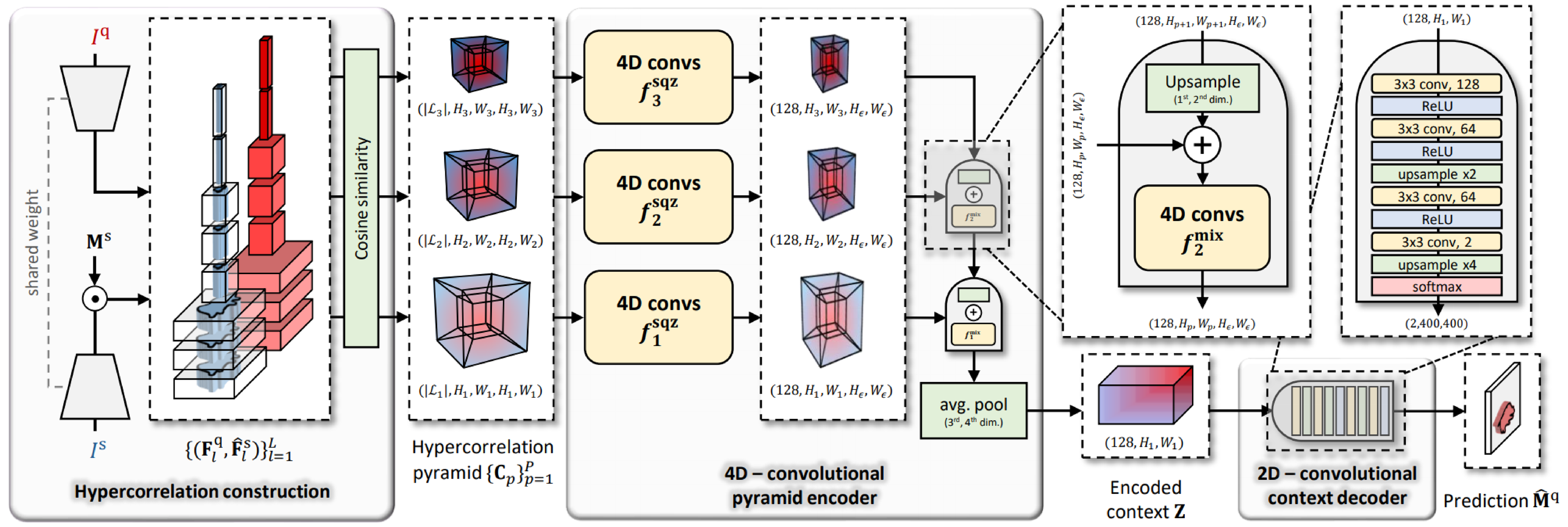
Proposed method

- **Hypercorrelation:** a form of "relational features" that represent relations between input images in multiple visual aspects, *i.e.*, multi-channel high-dimensional correlations

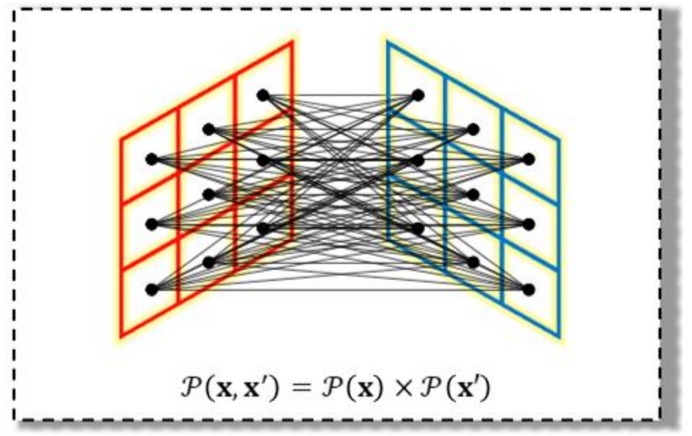
$$\hat{\mathbf{C}}_l(\mathbf{x}^{\mathrm{q}}, \mathbf{x}^{\mathrm{s}}) = \mathrm{ReLU}\left(\frac{\mathbf{F}^{\mathrm{q}}_l(\mathbf{x}^{\mathrm{q}}) \cdot \hat{\mathbf{F}}^{\mathrm{s}}_l(\mathbf{x}^{\mathrm{s}})}{\|\mathbf{F}^{\mathrm{q}}_l(\mathbf{x}^{\mathrm{q}})\|\|\hat{\mathbf{F}}^{\mathrm{s}}_l(\mathbf{x}^{\mathrm{s}})\|}\right) \qquad \mathbf{C}_p \in \mathbb{R}^{|\mathcal{L}_p| \times H_p \times W_p \times H_p \times W_p} \qquad \mathcal{C} = \{\mathbf{C}_p\}_{p=1}^{P}$$

4D correlation at each layer — Hypercorrelation at pyramid layer p — Hypercorrelation pyramid

- **Hypercorrelation Squeeze Networks:** captures relevant patterns in high-dim. correlations



- **Center-Pivot 4D Convolution:** a variant of 4D convolution for efficient correlation processing



4D convolutional kernel

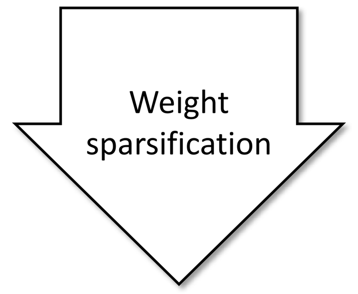
4D convolution

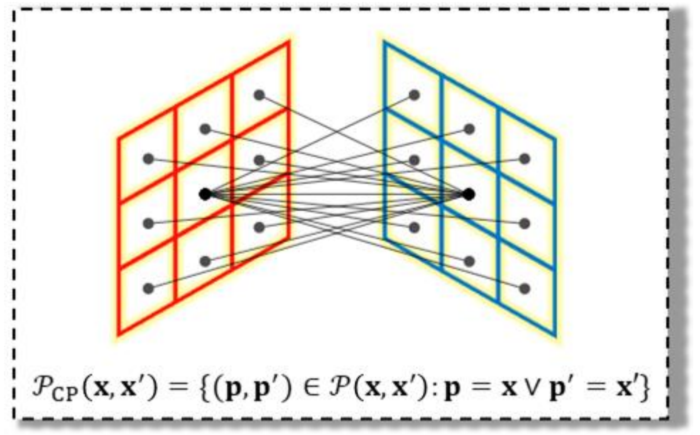
$$(c * k)(\mathbf{x}, \mathbf{x}') = \sum_{(\mathbf{p}, \mathbf{p}') \in \mathcal{P}(\mathbf{x}, \mathbf{x}')} c(\mathbf{p}, \mathbf{p}') k(\mathbf{p} - \mathbf{x}, \mathbf{p}' - \mathbf{x}')$$

1. Quadratic complexity with respect to the input feature maps
2. Over-parameterization of the high-dimensional convolutional kernel

From a set of neighborhood positions in a local 4D window, collect a small subset of activations that ***pivots*** either one of 2-dimensional ***centers***:

$$\mathcal{P}_{\mathrm{CP}}(\mathbf{x}, \mathbf{x}') = \{(\mathbf{p}, \mathbf{p}') \in \mathcal{P}(\mathbf{x}, \mathbf{x}') : \mathbf{p} = \mathbf{x} \lor \mathbf{p}' = \mathbf{x}'\}$$





Center-pivot 4D convolutional kernel

Center-pivot 4D convolution

$$(c * k_{\mathrm{CP}})(\mathbf{x}, \mathbf{x}') = \sum_{\mathbf{p}' \in \mathcal{P}(\mathbf{x}')} c(\mathbf{x}, \mathbf{p}') k_c^{2\mathrm{D}}(\mathbf{p}' - \mathbf{x}') + \sum_{\mathbf{p} \in \mathcal{P}(\mathbf{x})} c(\mathbf{p}, \mathbf{x}') k_{c'}^{2\mathrm{D}}(\mathbf{p} - \mathbf{x})$$

1. Reduced memory and time complexity: $\mathcal{O}(N^4) \rightarrow \boldsymbol{\mathcal{O}(N^2)}$
2. Reduced number of learnable parameters: 11.3M → **2.6M**

Experimental results and analyses

- **Evaluation results on standard few-shot segmentation datasets:**

PASCAL-5^i

Backbone network	Methods	1-shot 5^0	5^1	5^2	5^3	mean	FB-IoU	5-shot 5^0	5^1	5^2	5^3	mean	FB-IoU	# learnable params
ResNet50	PPNet (ECCV'20)	48.6	60.6	55.7	46.5	52.8	69.2	58.9	68.3	66.8	58.0	63.0	75.8	31.5M
	PFENet (TPAMI'20)	61.7	69.5	55.4	56.3	60.8	73.3	63.1	70.7	55.8	57.9	61.9	73.9	10.8M
	RePRI (CVPR'21)	59.8	68.3	**62.1**	48.5	59.7	-	64.6	71.4	**71.1**	59.3	66.6	-	-
	HSNet (ours)	**64.3**	**70.7**	60.3	**60.5**	**64.0**	**76.7**	**70.3**	**73.2**	67.4	**67.1**	**69.5**	**80.6**	**2.6M**
ResNet101	PPNet (ECCV'20)	52.7	62.8	57.4	47.7	55.2	70.9	60.3	70.0	**69.4**	60.7	65.1	77.5	50.5M
	PFENet (TPAMI'20)	60.5	69.4	54.4	55.9	60.1	72.9	62.8	70.4	54.9	57.6	61.4	73.5	10.8M
	RePRI (CVPR'21)	59.6	68.6	**62.2**	47.2	59.4	-	66.2	71.4	67.0	57.7	65.6	-	-
	HSNet (ours)	**67.3**	**72.3**	62.0	**63.1**	**66.2**	**77.6**	**71.8**	**74.4**	67.0	**68.3**	**70.4**	**80.6**	**2.6M**
	HSNet† (ours)	66.2	69.5	53.9	56.2	61.5	72.5	68.9	71.9	56.3	57.9	63.7	73.8	**2.6M**

COCO-20^i

Backbone network	Methods	1-shot 20^0	20^1	20^2	20^3	mean	5-shot 20^0	20^1	20^2	20^3	mean
ResNet50	RPMM (ECCV'20)	29.5	36.8	28.9	27.0	30.6	33.8	42.0	33.0	33.3	35.5
	PFENet (TPAMI'20)	**36.5**	38.6	34.5	33.8	35.8	36.5	43.3	37.8	38.4	39.0
	RePRI (CVPR'21)	32.0	38.7	32.7	33.1	34.1	39.3	45.4	39.7	41.8	41.6
	HSNet (ours)	36.3	**43.1**	**38.7**	**38.7**	**39.2**	**43.3**	**51.3**	**48.2**	**45.0**	**46.9**
ResNet101	DAN (ECCV'20)	-	-	-	-	24.4	-	-	-	-	29.6
	PFENet (TPAMI'21)	36.8	41.8	38.7	36.7	38.5	40.4	46.8	43.2	40.5	65.8
	HSNet (ours)	**37.2**	**44.1**	**42.4**	**41.3**	**41.2**	**45.9**	**53.0**	**51.8**	**47.1**	**49.5**

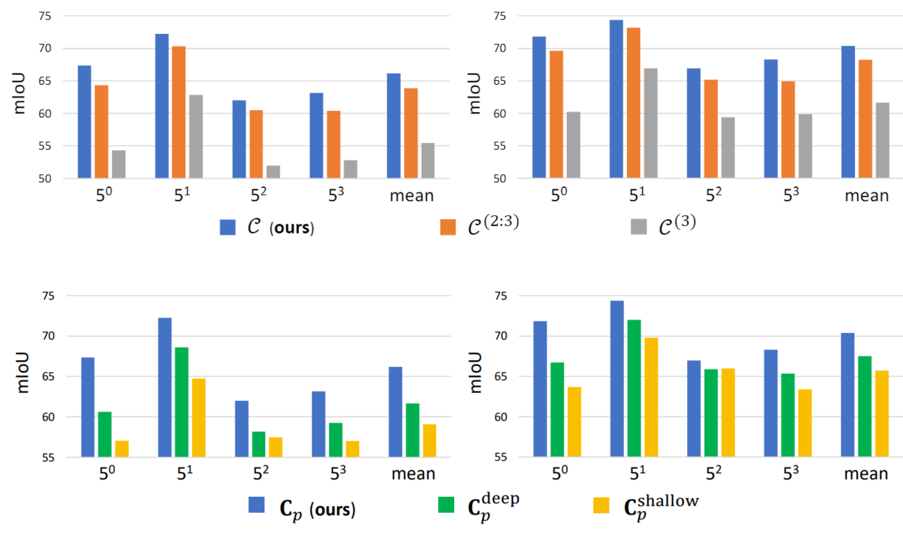
FSS-1000

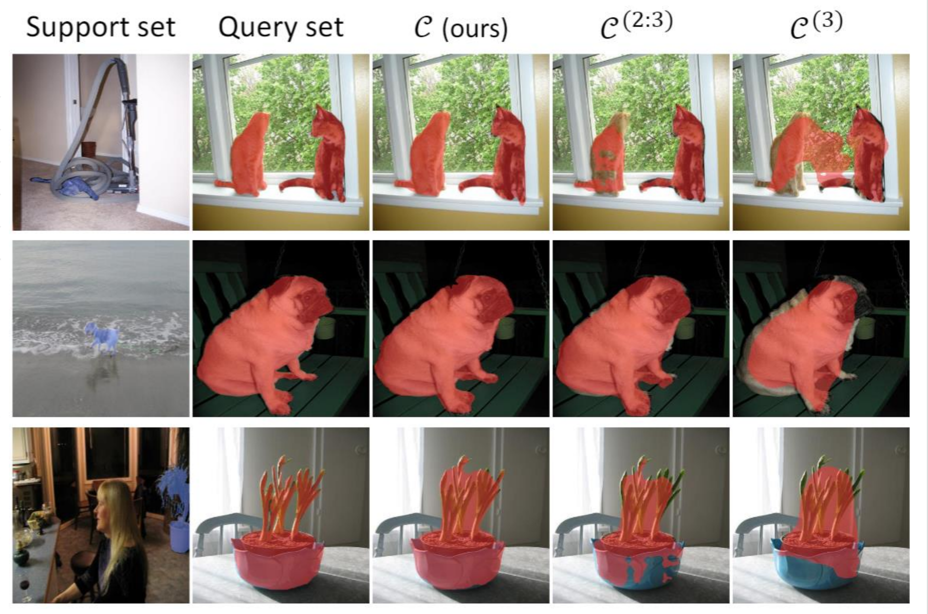
Backbone network	Methods	mIoU 1-shot	mIoU 5-shot
VGG16	OSLSM (BMVC'17)	70.3	73.0
	FSS (CVPR'20)	73.5	80.1
	DoG-LSTM (WACV'21)	80.8	83.4
	HSNet (ours)	**82.3**	**85.8**
ResNet50	HSNet (ours)	**85.5**	**87.8**
ResNet101	DAN (ECCV'20)	85.2	88.1
	HSNet (ours)	**86.5**	**88.5**

- **Effectiveness of center-pivot 4D kernel:**

Kernel type	1-shot 5^0	5^1	5^2	5^3	mean	5-shot 5^0	5^1	5^2	5^3	mean	# learnable params	time (*ms*)	memory footprint (*GB*)	FLOPs (*G*)
Original 4D kernel (NeurIPS'18)	64.5	71.4	62.3	61.7	64.9	70.8	**74.8**	**67.4**	67.5	70.1	11.3M	512.17	4.12	702.35
Separable 4D kernel (NeurIPS'19)	66.1	72.0	**63.2**	62.6	65.9	71.2	74.1	67.2	68.1	70.2	4.4M	28.48	1.50	28.40
Center-pivot 4D kernel (ours)	**67.3**	**72.3**	62.0	**63.1**	**66.2**	**71.8**	74.4	67.0	**68.3**	**70.4**	**2.6M**	**25.51**	**1.39**	**20.56**

- **Effect of hypercorrelations:**





- **Robustness to domain shift:**

Evaluation results of COCO-20^i-trained model on each fold of PASCAL-5^i

Method	COCO→PASCAL 1-shot	COCO→PASCAL 5-shot	# params to train	data augmentation used during training
PFENet$_{\mathrm{res50}}$ (TPAMI'20)	61.1	63.4	10.8M	flip, rotate, crop
RePRI$_{\mathrm{res50}}$ (CVPR'21)	**63.2**	67.7	46.7M	flip
HSNet$_{\mathrm{res50}}$(ours)	61.6	**68.7**	**2.6M**	**none**
HSNet$_{\mathrm{res101}}$(ours)	**64.1**	**70.3**	**2.6M**	**none**

- **Effect of finetuning:**