Convolutional Hough Matching Networks

Juhong Min Minsu Cho
POSTECH CSE & GSAI

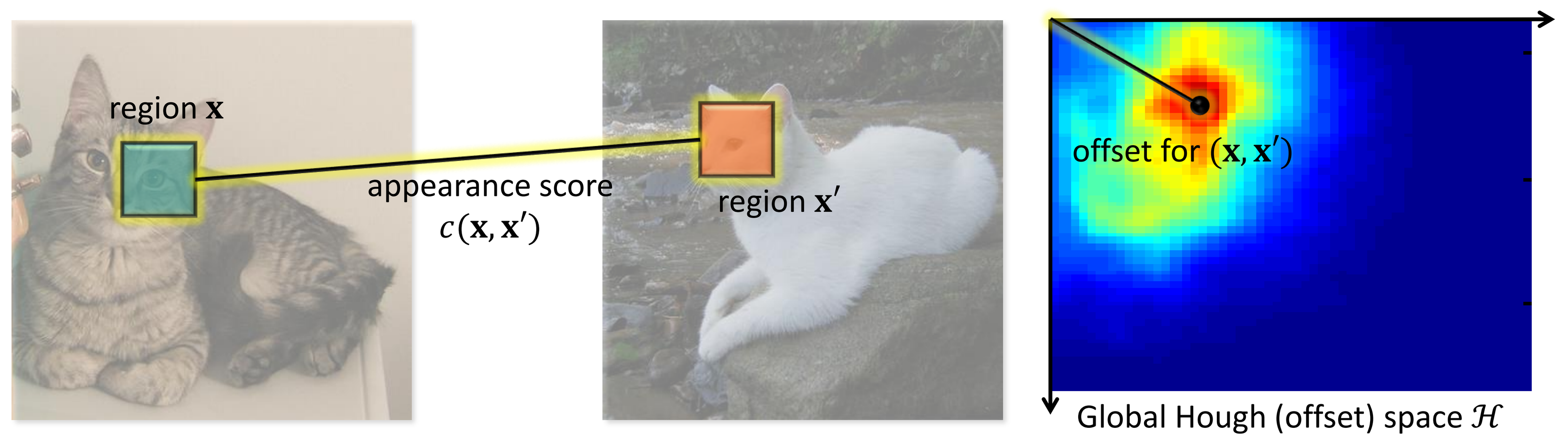
project page arXiv code

Problem Introduction & Contribution

- **Semantic visual correspondence:**
  - Matching images depicting different instances of the same object class
  - Visual correspondence "in the wild," generalizing to other matching tasks
  - Core component for many vision tasks, *e.g.*, tracking, retrieval, etc.
- **Our contributions:**
  - Introduce a **Hough transform perspective** on convolutional matching
  - Develop trainable CHM layer with **semi-isotropic high-dimensional** kernel
  - Propose CHMNet with a **small number of interpretable parameters**
  - **SOTA** on three standard benchmarks of semantic correspondence

Hough Matching (HM)

Hough matching is the algorithm of *Cho et al*. (CVPR 2015) which reweights appearance similarity by Hough `voting' to enforce geometric consistency



$$v(\mathbf{h}) = \sum_{(\mathbf{x},\mathbf{x}') \in \mathcal{X}\times\mathcal{X}'} c(\mathbf{x},\mathbf{x}') k_{\text{iso}}(\|(\mathbf{x}'-\mathbf{x}) - \mathbf{h}\|_g)$$

Appearance score, *e.g.*, cosine similarity

Kernel that assigns a voting score according to how close $(\mathbf{x}'-\mathbf{x})$ is to $\mathbf{h}$

Distance between observed offset $(\mathbf{x}'-\mathbf{x})$ and the given offset $\mathbf{h}$ in the Hough space

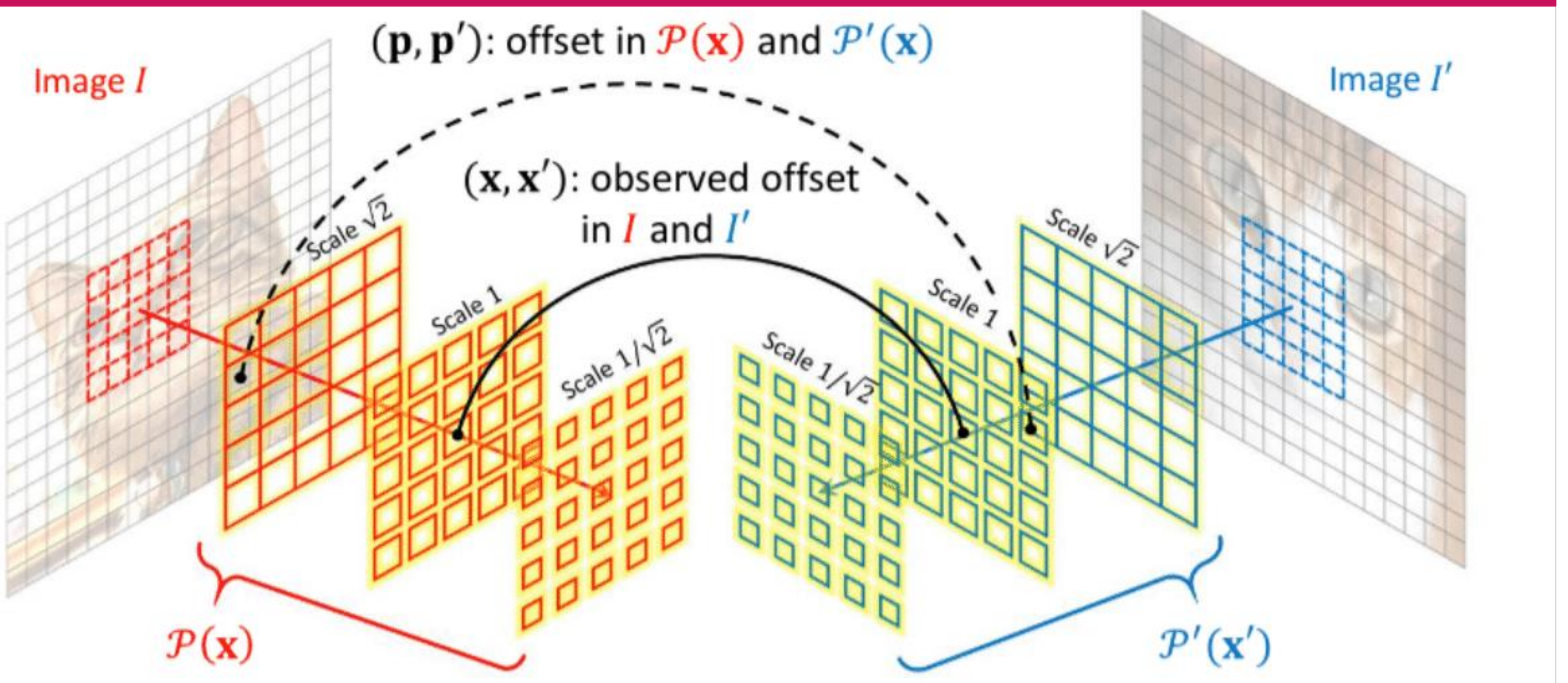
- **Limitation:** The Hough space is shared for all candidate matches so it cannot capture the reliability of a specific candidate matches, thus being less accurate and weak to background clutters.
- **Solution:** to create a **local** & **individual** voting space for each match, *i.e.*, convolutionalization of the Hough matching algorithm.

Convolutional Hough Matching (CHM)

(global & shared) (local & individual)

HM → CHM

1. We introduce local windows, $\mathcal{P}(\mathbf{x})$ and $\mathcal{P}'(\mathbf{x}')$, around regions $\mathbf{x}$ and $\mathbf{x}'$.
2. The local voting space is now dedicated to $(\mathbf{x}, \mathbf{x}')$.
3. Let $k(\mathbf{z}, \mathbf{z}')$ represent kernel value corresponding to two positions, $\mathbf{z}$ and $\mathbf{z}'$.

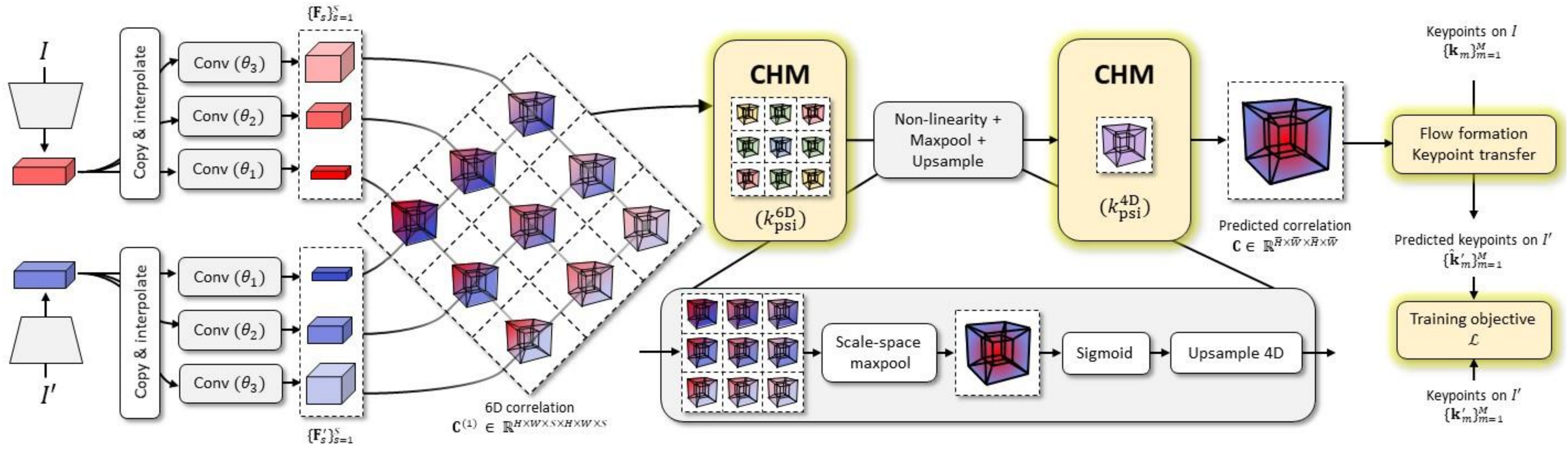


4. Then we have, $c_{\text{HM}}(\mathbf{x}, \mathbf{x}') = \sum_{(\mathbf{p},\mathbf{p}') \in \mathcal{P}(\mathbf{x})\times\mathcal{P}'(\mathbf{x}')} c(\mathbf{p},\mathbf{p}') k(\mathbf{p}-\mathbf{x}, \mathbf{p}'-\mathbf{x}')$ (We refer the readers to the paper for a complete derivation of CHM)

High-dimensional convolution on correlation tensor as local & individual Hough matching

We further relax isotropy and propose position-sensitive isotropic kernel k_{psi} which shares parameters whose triplets $(\|\mathbf{p}'-\mathbf{p}\|_g, \|\mathbf{p}-\mathbf{x}\|_g, \|\mathbf{p}'-\mathbf{x}'\|_g)$ are the same.

- **Advantage of CHM over existing 4D convolutions on correlation:**
  1. **Generalizability**: voting space can be extended to higher dim. beyond 4D, *e.g.*, 6D (translation & scale)
  2. **Scalability**: channel size of 1 & parameter sharing → small number of parameters
  3. **Interpretability**: a single kernel for each layer eases kernel visualization
  4. **Performance**: state-of-the-art performance on three standard benchmark datasets
- **Convolutional Hough matching networks** (The CHM part with k_{psi} has 275 parameters):



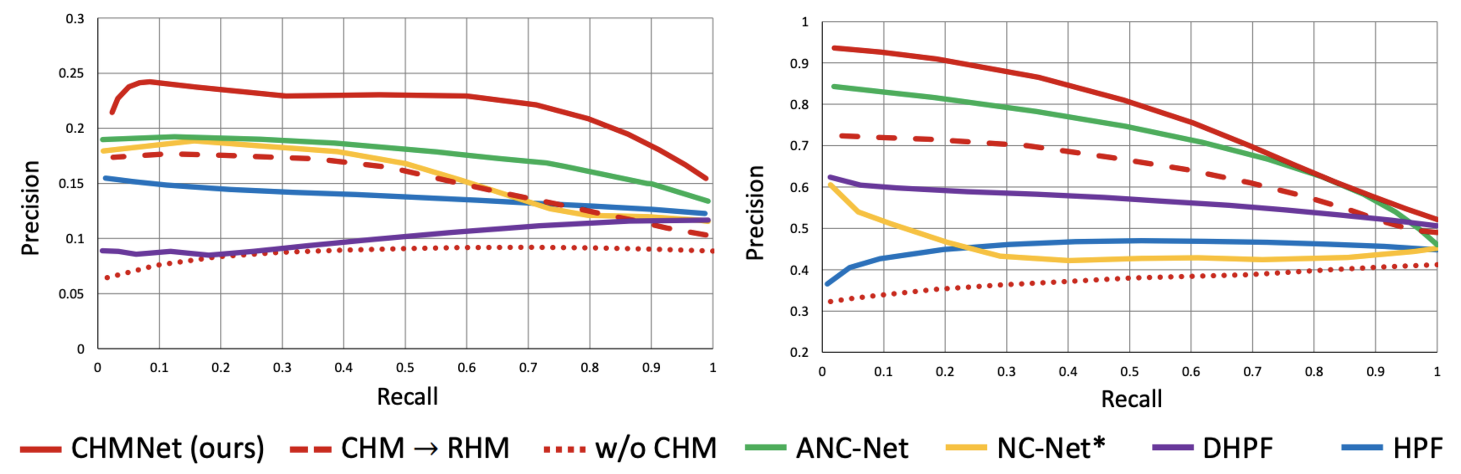
- **Flow formation & keypoint transfer:** soft-argmax with soft sampler
- **Objective:** minimizes L2-distance between predicted and GT keypoints: $\mathcal{L} = \frac{1}{M}\sum_{m=1}^{M} \|\hat{\mathbf{k}}'_m - \mathbf{k}'_m\|$

Experimental Results and Analyses

- **Evaluation results on semantic correspondence benchmarks:**

Methods	SPair-71k PCK @ α_{bbox} 0.1 (F)	SPair-71k PCK @ α_{bbox} 0.1 (T)	PF-PASCAL PCK @ α_{img} 0.05	PF-PASCAL PCK @ α_{img} 0.1	PF-WILLOW PCK @ α_{bbox} 0.05	PF-WILLOW PCK @ α_{bbox} 0.1	uses nD conv?	FLOPs (G)	time (*ms*)	memory (GB)
$\text{UCN}_{\text{res101}}$ (NeurIPS'16)	-	17.7	-	75.1	-	-	✗	-	-	-
$\text{HPF}_{\text{res101}}$ (ICCV'19)	28.2	-	60.1	84.8	45.9	74.4	✗	-	63	-
$\text{SCOT}_{\text{res101}}$ (CVPR'20)	35.6	-	63.1	85.4	47.8	76.0	✗	6.2	151	4.6
$\text{DHPF}_{\text{res101}}$ (ECCV'20)	37.3	27.4	75.7	90.7	49.5	77.6	✗	**2.0**	58	1.6
$\text{NC-Net}^{*}_{\text{res101}}$ (NeurIPS'18)	-	-	-	81.9	-	-	4D	44.9	222	1.2
$\text{DCC-Net}^{*}_{\text{res101}}$ (ICCV'19)	-	-	-	83.7	-	-	4D	47.1	567	2.7
$\text{ANC-Net}_{\text{res101}}$ (CVPR'20)	-	28.7	-	86.1	-	-	4D	44.9	216	**0.9**
$\text{CHMNet}_{\text{res101}}$ (ours)	**46.3**	**30.1**	**80.1**	**91.6**	**52.7**	79.4	6D	19.6	**54**† (248)	1.6

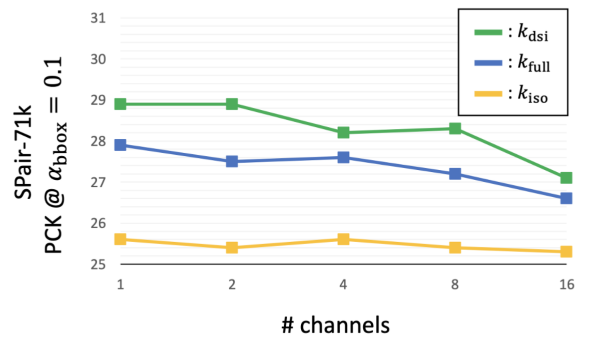
PR curves on SPair-71k PR curves on PF-PASCAL



$\text{Precision} = \frac{N_{\text{TP}}}{N_{\text{TP}}+N_{\text{FP}}}$

$\text{Recall} = \frac{N_{\text{TP}}}{N_{\text{mask}}}$

- **Channel size experiments:** High-dimensional convolution on a correlation tensor is to learn a *reliable voting strategy* rather than to capture diverse patterns in the correlation tensor.



- **Qualitative results & learned kernel (k_{psi}) visualization:**