UNIVERSITY OF MICHIGAN

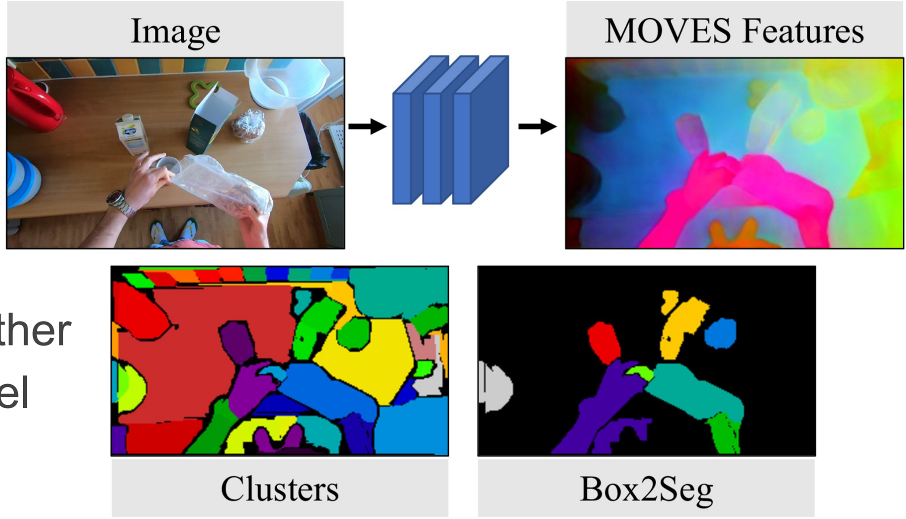
JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

MOVES: Manipulated Objects in Video Enable Segmentation

Richard E.L. Higgins and David F. Fouhey
University of Michigan
{relh, fouhey}@umich.edu

MOVES 1-minute Overview

- People moving objects creates motion that disagrees with camera motion
- We use motion cues to teach a network:
  - **Grouping:** do pixel X and pixel Y go together
  - **Association:** is this hand holding that pixel
- At training time, we use epipolar geometry and optical flow to train with grouping/association pseudolabels
- At inference time, we produce pixel-wise embeddings and held-object association maps from a single RGB image



Teaser

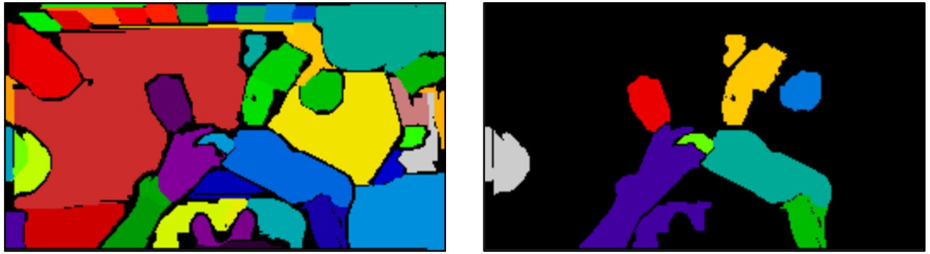
- Here we can see someone making breakfast, we:
  - Understand the bag is an object
  - Recognize a hand is holding it
  - Understand background objects

Teaser

- Here we can see someone making breakfast, we:
  - Understand the bag is an object
  - Recognize a hand is holding it
  - Understand background objects

- We want a system that does the same:
  - Groups held objects
  - Recognizes contact between hands and objects
  - Groups non-held objects

- Discriminative training on simple pseudolabels works pretty well

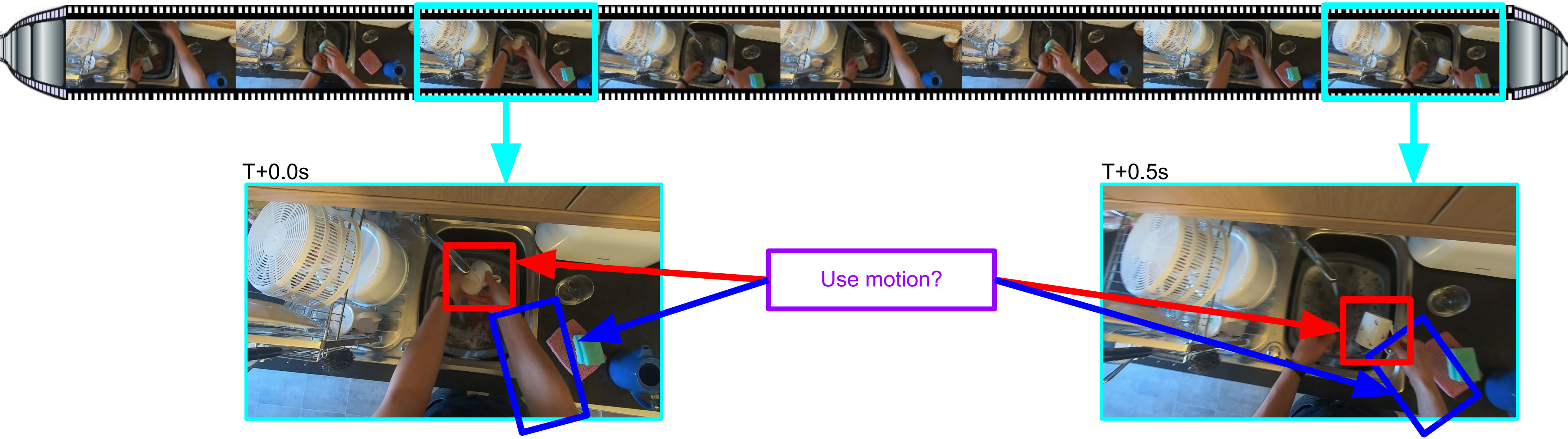


Clusters

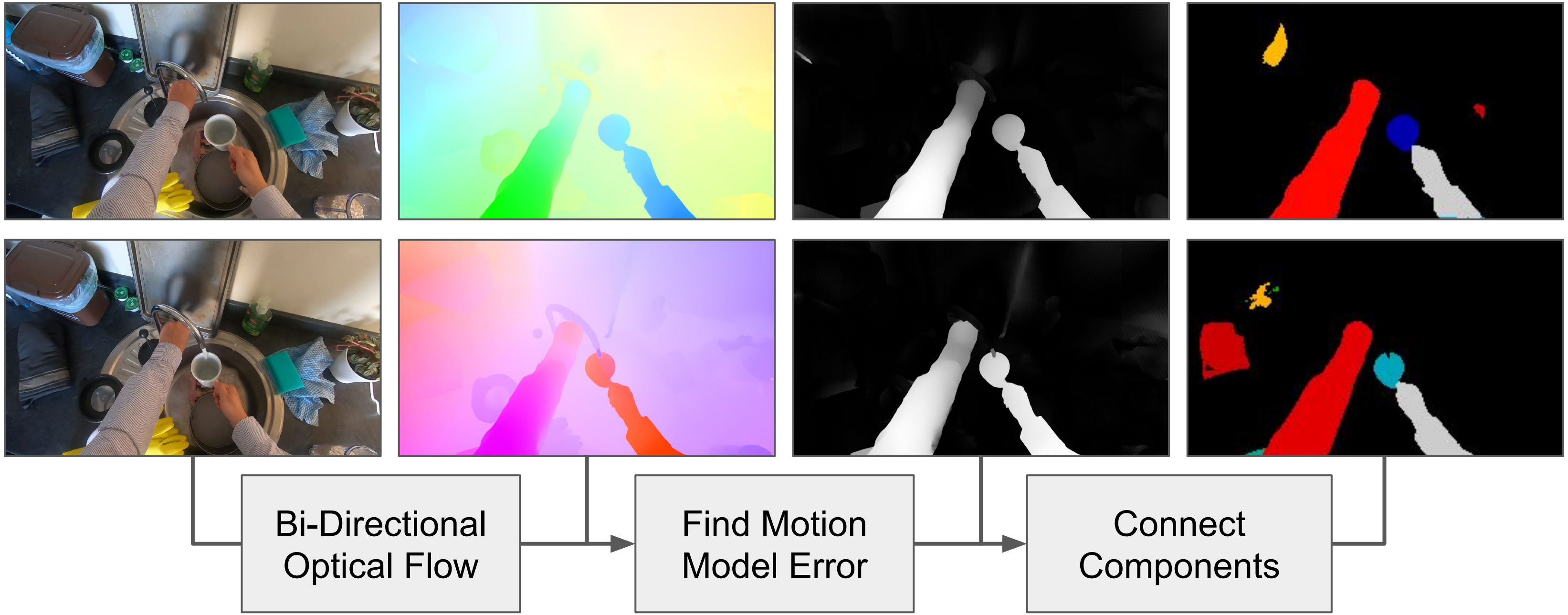
Box2Seg

Problem

How can we use motion cues generated by manipulation to segment and associate hands and held-objects in egocentric video?



Pseudo-labels Pt 1.

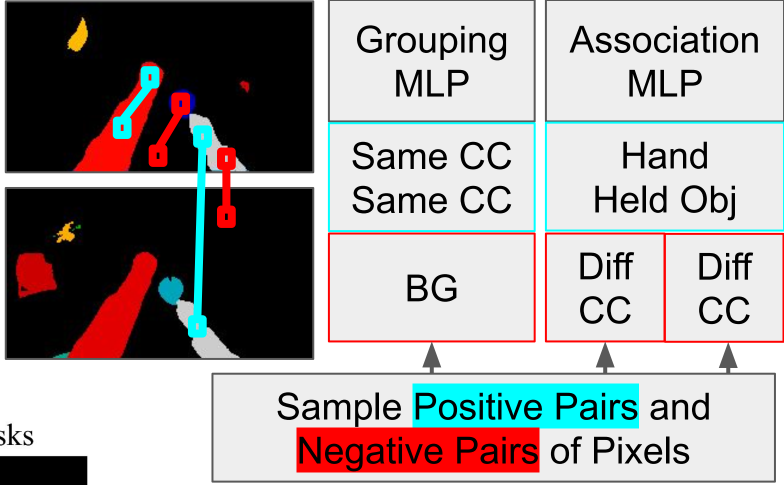


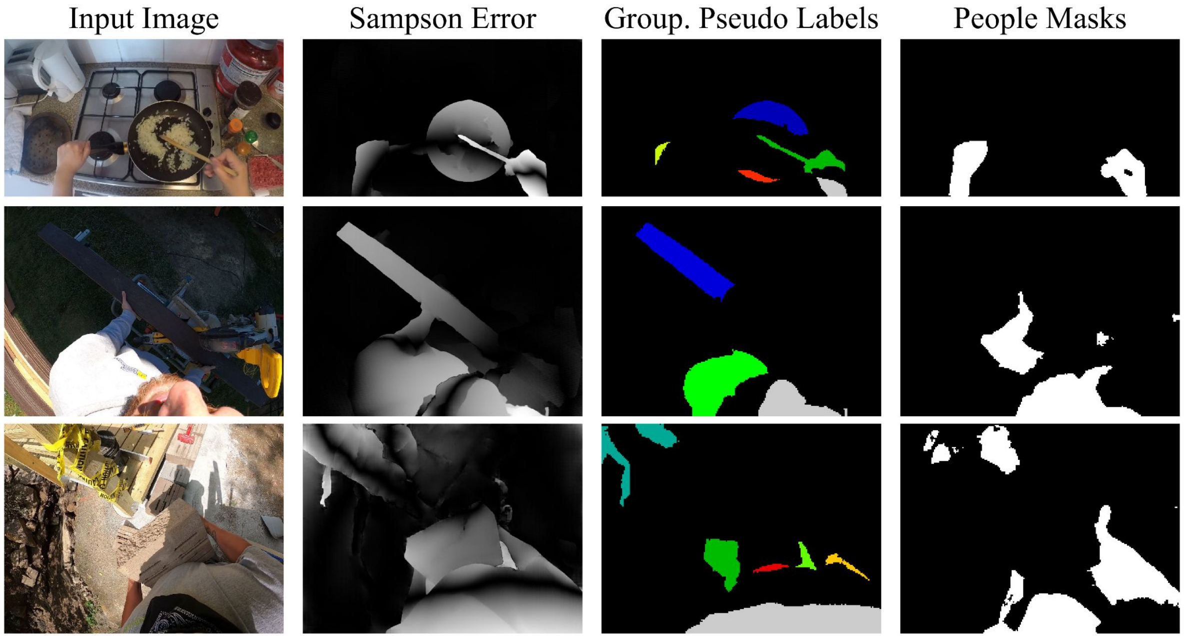
1. Choose video frames offset ~0.5s
2. Run bi-directional optical flow
3. Find cyclic correspondences in this flow
4. Use RANSAC to estimate a fundamental matrix between the two frames
5. Ideally, person and held-object motion are outliers, disagreeing with this motion model
6. Calculate sampson epipolar error for all correspondences
7. Run connected components on error regions above threshold

Pseudo-labels Pt 2.

Grouping. $\mathbf{G}_{i,j}$ is: positive if i, j are in the same foreground connected component; negative if i is in the foreground and j is not; and unknown otherwise.

Hand Association. We use the the Ternaus [19] person binary segmentation system, assuming the data is egocentric and so the visible people are hands. The association $\mathbf{A}_{i,j}$ is: positive if i, j are in the same connected component and have differing person predictions; negative if i, j are in different components; and unknown otherwise.





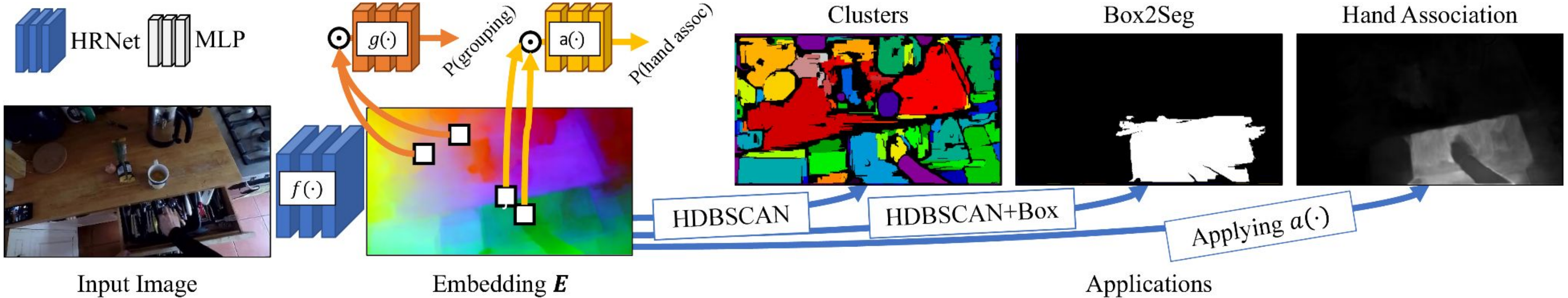
Method – Training

At training time, we assume an image $\mathbf{I}$ and pseudolabels for grouping $\mathbf{G}$ and association $\mathbf{A}$ that identify each pair of pixels i and j as either positive (e.g., $\mathbf{G}_{i,j} = 1$), negative ($\mathbf{G}_{i,j} = -1$), or unknown ($\mathbf{G}_{i,j} = 0$) and similarly for $\mathbf{A}$. Given a set $\mathcal{S}$ of pairs of pixels, we directly minimize the binary cross-entropy loss (denoted $\mathrm{CE}(y, \hat{y})$) applied to the classification head outputs, or:

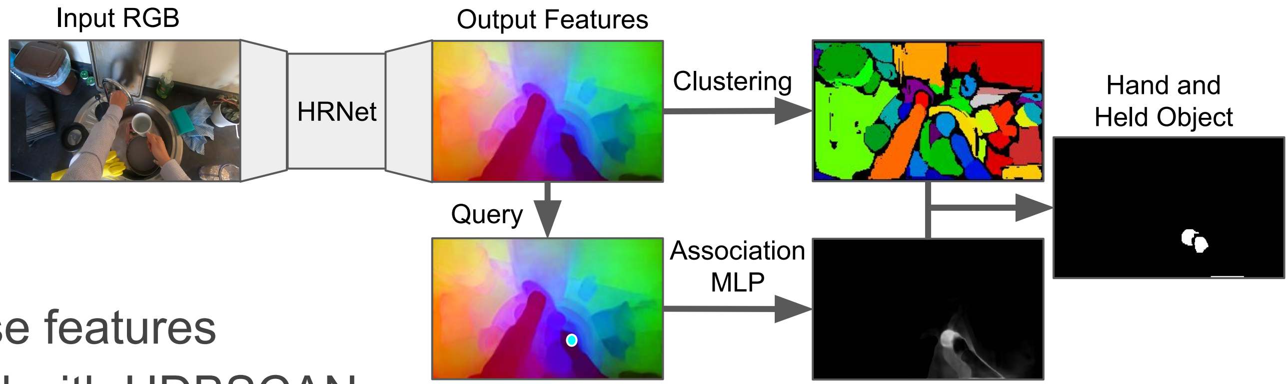
$$\frac{W}{|\mathcal{S}|} \sum_{(i,j)\in\mathcal{S}} \mathrm{CE}(\mathbf{G}_{i,j}, g(\mathbf{e}_{i,j})) + \mathrm{CE}(\mathbf{A}_{i,j}, a(\mathbf{e}_{i,j})) \quad (1)$$

where $\mathbf{e}_{i,j} \in \mathbb{R}^{2F}$ is defined as the concatenation of the ith pixel and jth pixel of $\mathbf{E} = f(\mathbf{I})$ (i.e., $\mathbf{e}_{i,j} = [\mathbf{E}[i], \mathbf{E}[j]]$)

- Dense embeddings are learnt only from sampling pairs of pixels to use with the grouping and association MLPs.
- HDBSCAN clustering reveals objects, and we segment held-objects using hand query points and the association MLP.



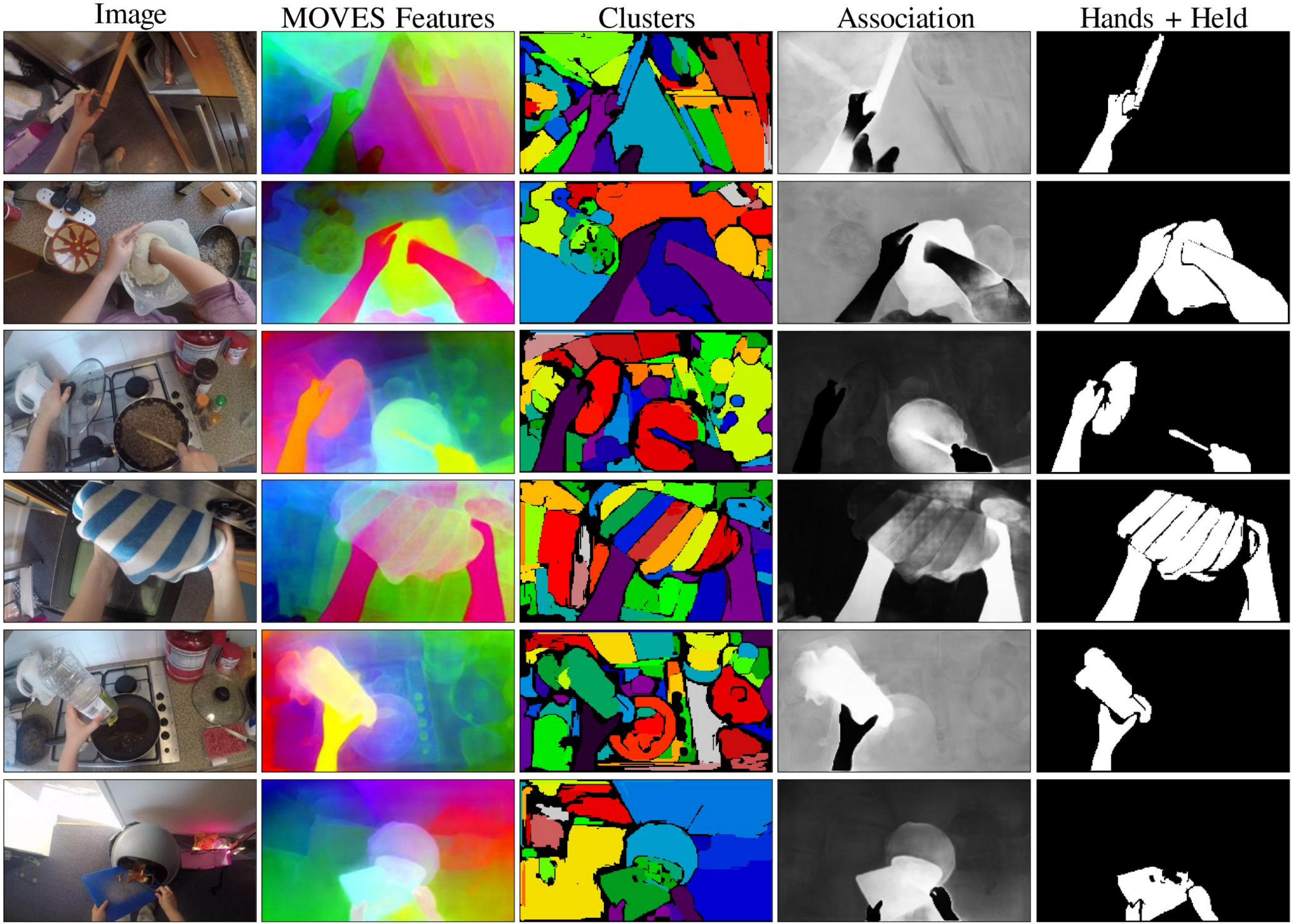
Method – Inference



1. HRNet produces dense features
2. Features are clustered with HDBSCAN
3. A query point on a hand is input
4. Association is predicted for this hand
5. Association is averaged within clusters
6. Association above a threshold becomes a held-object

Results -

Epic Kitchens



Results -

Epic Kitchens

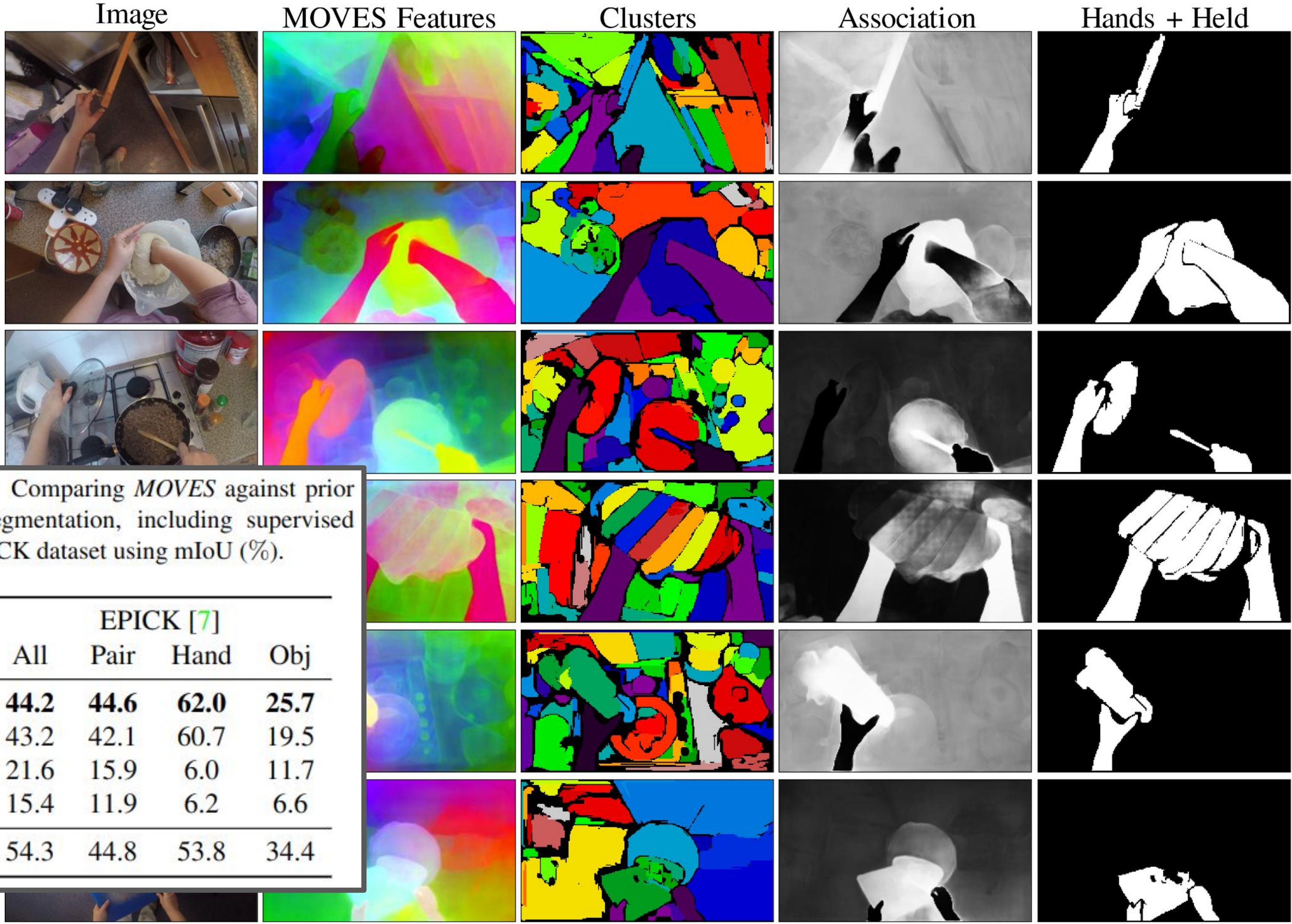
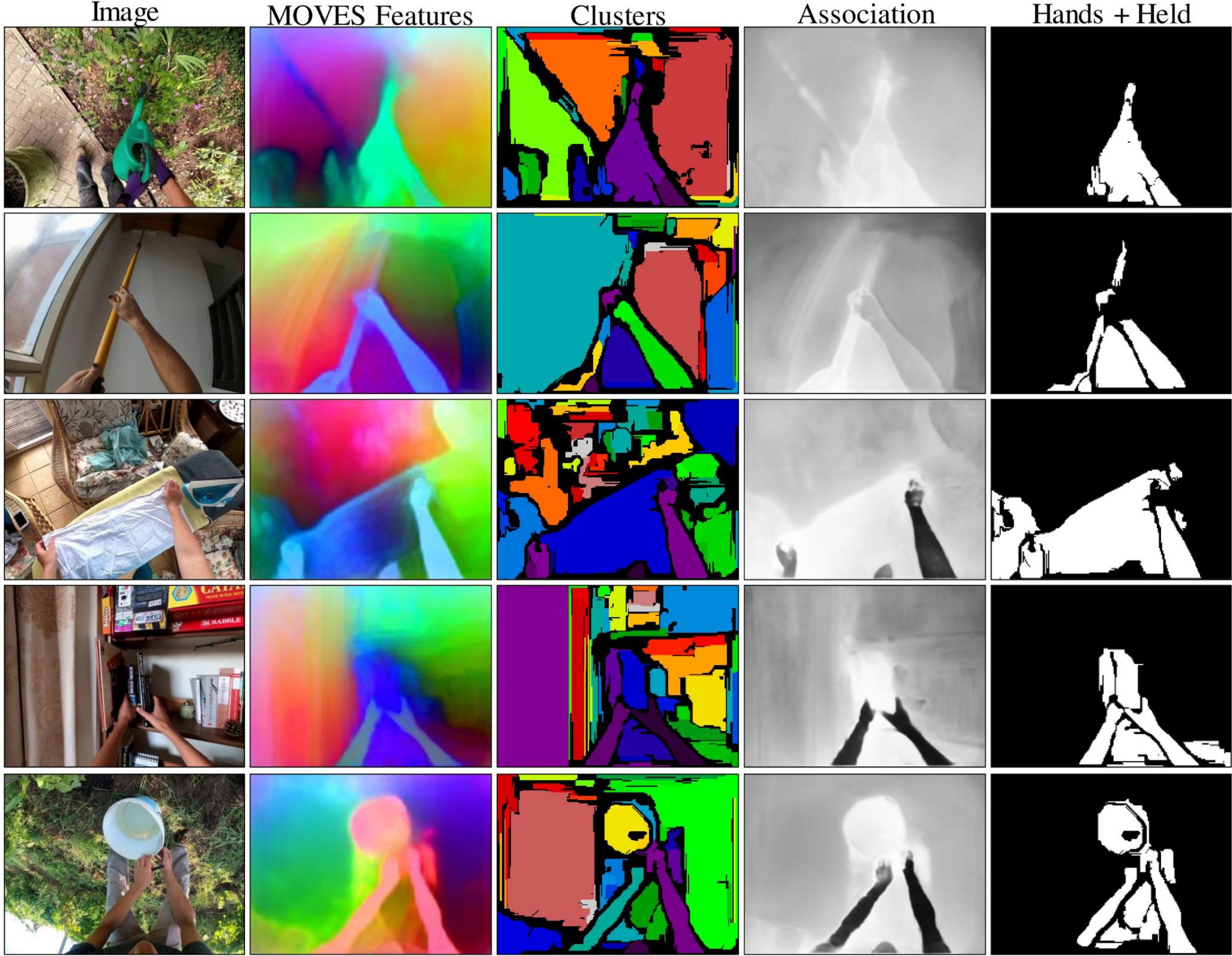


Table 1. **HOS Performance.** Comparing *MOVES* against prior methods on Hand+Object Segmentation, including supervised methods, evaluated on the EPICK dataset using mIoU (%).

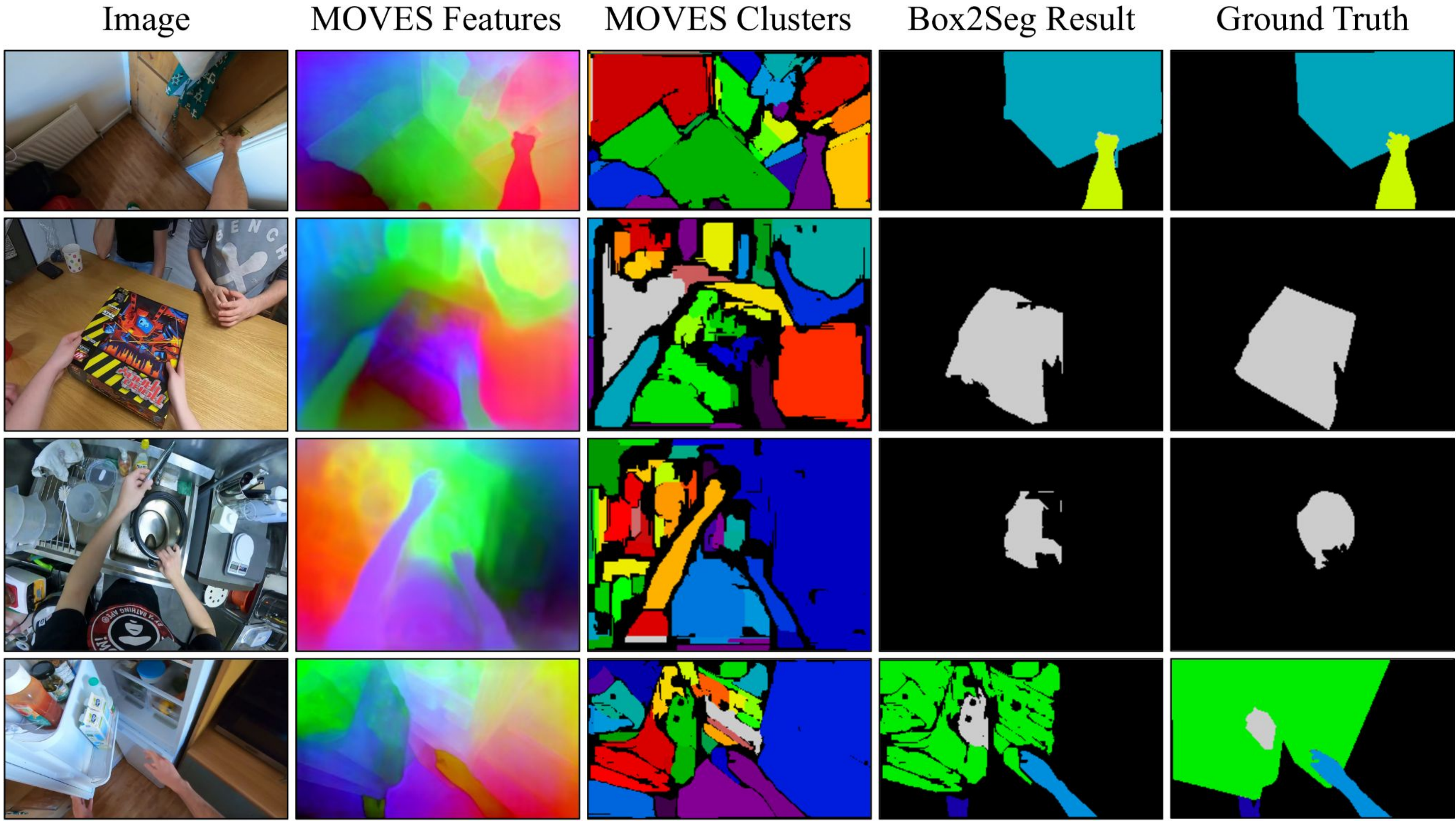
	EPICK [7]			
	All	Pair	Hand	Obj
MOVES	**44.2**	**44.6**	**62.0**	**25.7**
COHESIV [37]	43.2	42.1	60.7	19.5
Saliency [48]	21.6	15.9	6.0	11.7
Flow [40]	15.4	11.9	6.2	6.6
Supervised BBox [36]	54.3	44.8	53.8	34.4

Results -

Ego4D



Results - Box2Seg



Results - Box2Seg

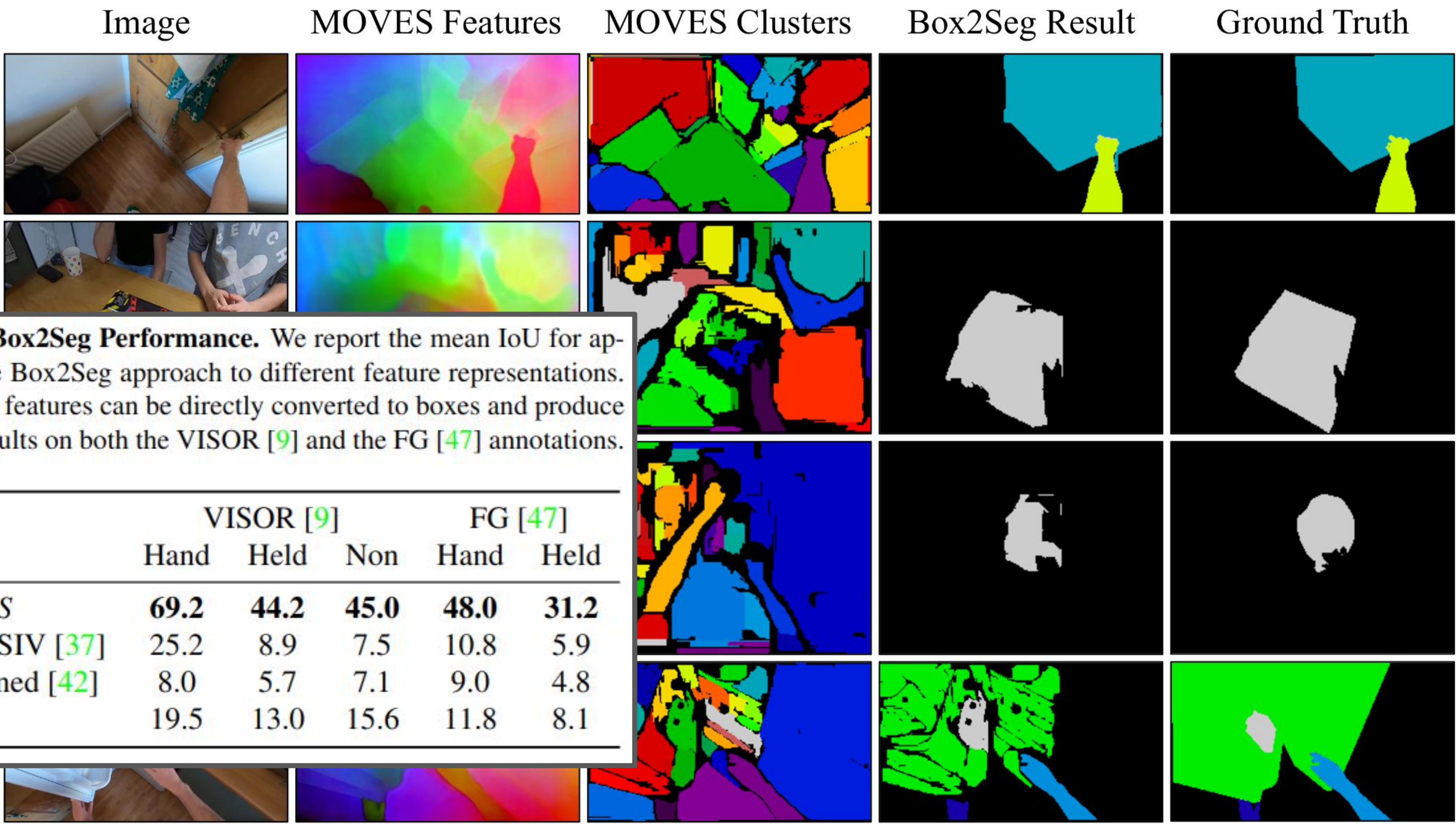
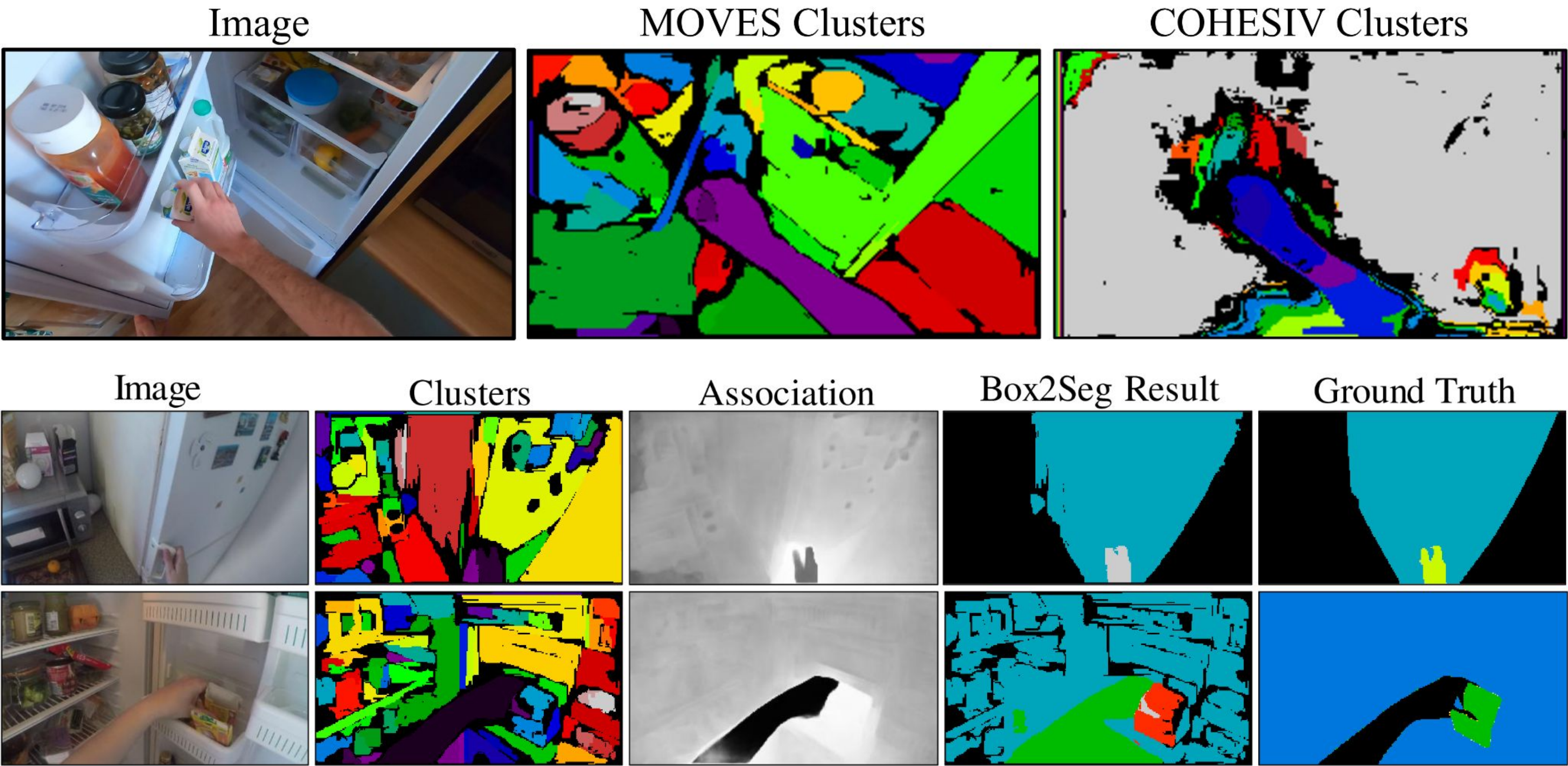


Table 2. **Box2Seg Performance.** We report the mean IoU for applying the Box2Seg approach to different feature representations. *MOVES*'s features can be directly converted to boxes and produce strong results on both the VISOR [9] and the FG [47] annotations.

	VISOR [9]			FG [47]	
	Hand	Held	Non	Hand	Held
MOVES	**69.2**	**44.2**	**45.0**	**48.0**	**31.2**
COHESIV [37]	25.2	8.9	7.5	10.8	5.9
Pretrained [42]	8.0	5.7	7.1	9.0	4.8
RGB	19.5	13.0	15.6	11.8	8.1

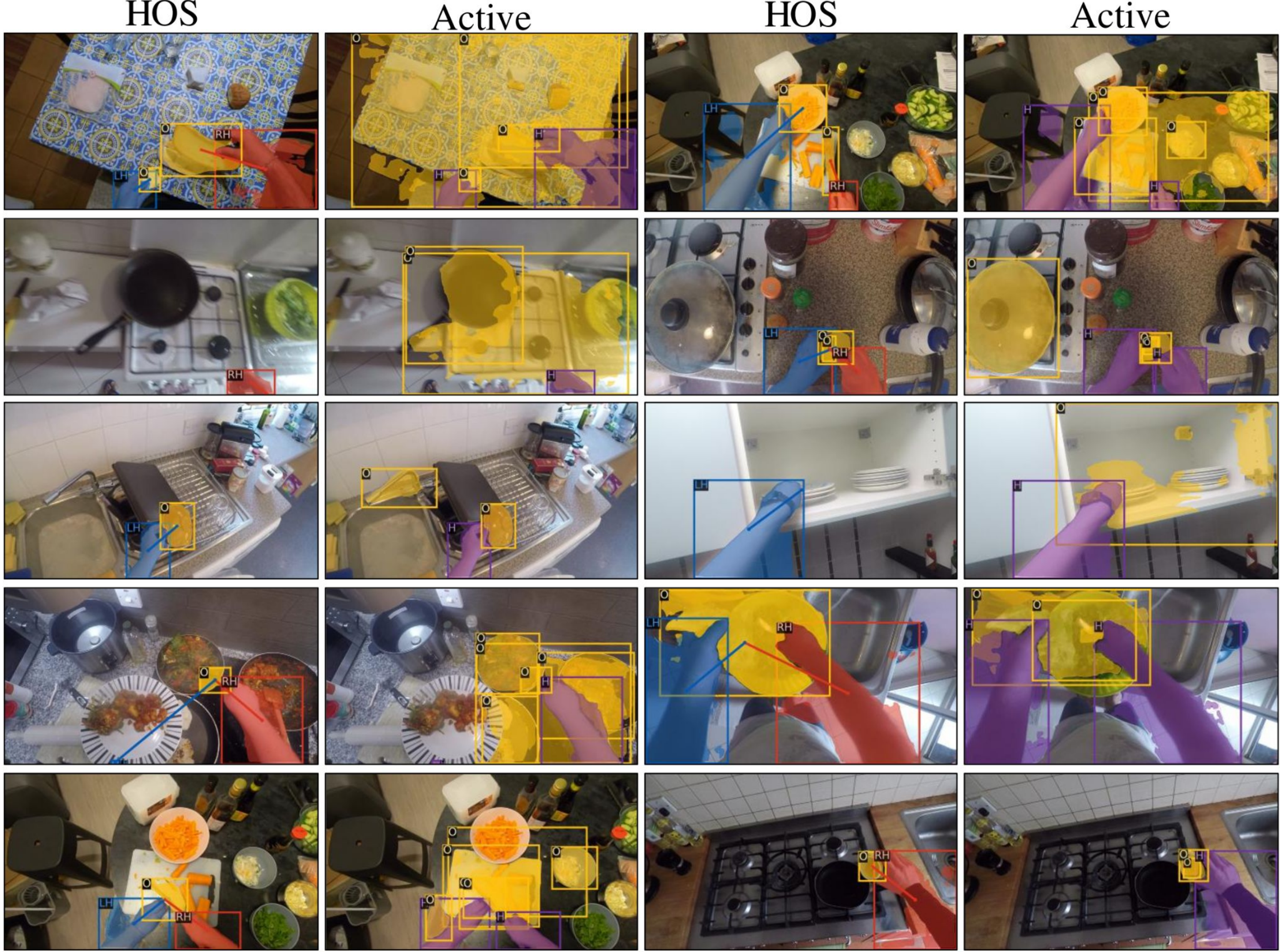
Results - Fridges (and comparison to COHESIV)



Results -

Training Label Generation…

Then training PointRend on it!



Comparisons - DINO, Playroom, New Labels, PWC Flow

Table 1. Additional AUROC Evaluations on VISOR.

	Hand	Held	Non
MOVES	**99.5**	**95.2**	**95.0**
MOVES (Alt Pseudolabels)	98.9	94.4	94.5
MOVES (PWC Optical Flow)	98.2	94.6	94.2
DINO	93.1	91.1	89.3

Table 2. Additional Evaluations on Playroom.

	Train Set	val (mIoU)
EISEN	Playroom	**73.0**
MOVES	Playroom	71.8

- Better grouping than DINO
- Out-of-the-box comparable to EISEN
- Pseudolabels optical flow > X works
- Pseudolabels using PWC flow works

Conclusions

- MOVES shows that simple learning signals and architectural design can lead to effective grouping and association.

- While our auto generated pseudo-labels are grossly inadequate in any one image, using them to train a network on thousands of images at scale leads to effective features.

Future Work

- MOVES shows that a network can implicitly embed association, such that comparing pairs of feature vectors works well. But association could be better modelled explicitly, with querying and attention.

- HDBSCAN and clustering in general can probably factor into both training and inference for more models with dense features.. RAPIDS.ai has clustering faster than neural network inference.