MOVES: Manipulated Objects in Video Enable Segmentation

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

Abstract

Our method uses manipulation in video to learn to understand held-objects and hand-object contact. We train a system that takes a single RGB image and produces a pixel-embedding that can be used to answer grouping questions (do these two pixels go together) as well as hand-association questions (is this hand holding that pixel). Rather than painstakingly annotate segmentation masks, we observe people in realistic video data. We show that pairing epipolar geometry with modern optical flow produces simple and effective pseudo-labels for grouping. Given people segmentations, we can further associate pixels with hands to understand contact. Our system achieves competitive results on hand and hand-held object tasks.

1. Introduction

Fig. 1 shows someone making breakfast. Despite having never been there, you understand the bag the hand is holding as an object, recognize that the hand is holding the bag, and recognize that the milk carton in the background is a distinct object. The goal of this paper is to build a computer vision system with such capabilities: grouping held objects (the bag), recognizing contact (the hand holding the bag), and grouping non-held objects (the carton). We accomplish our aim by pairing modern optical flow with 3D geometry and, to associate objects with hands, per-pixel human masks. Our results show that direct discriminative training on simple pseudo-labels generated by epipolar geometry produces strong feature representations that we can use to solve a variety of hand-held object-related tasks.

The topic of understanding hands and the objects they hold has been a subject of intense interest from the computer vision community for decades. Recently, this has often taken the form of extensive efforts annotating hands and hand-held objects [9, 13, 36, 47]. These methods often go beyond standard detection and segmentation approaches [17, 25] by producing associations between hands and objects and by detecting on any *held* object, as opposed to a fixed set of pre-defined object classes. Since these require expensive annotations, many researchers have started focusing on using weaker supervision [12, 37] by starting with a few readily obtained cues (e.g., basic information about humans, flow). These weakly-supervised methods, however, have not matched supervised methods regardless of supervision, methods like [36, 37] only understand objects when they are held and cannot group un-held objects.

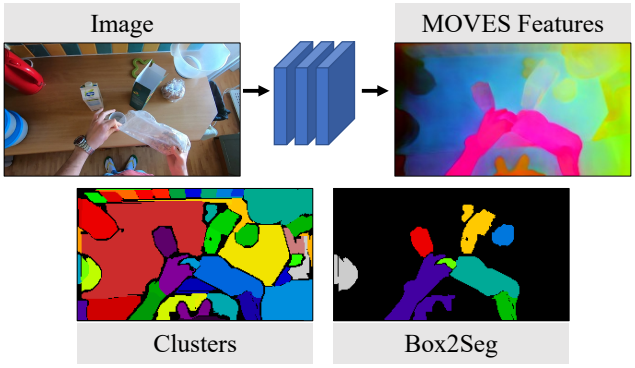

Figure 1. Given an input image, *MOVES* produces features (shown using PCA to project to RGB) that easily group with ordinary clustering systems and can also be used to associate hands with the objects they hold. The clusters are often sufficient for defining objects, but additional cues such as a box further improve them. At training time, *MOVES* learns this feature space from direct discriminative training on simple pseudo-labels. While *MOVES* learns only from objects that hands are actively holding (such as the semi-transparent bag), we show that it works well on inactive objects as well (such as the milk carton).

We propose a simple approach based on directly predicting two properties: *grouping*, or whether pixels move together (the classic Gestalt law of common fate [43]); as well as *hand association*, whether a hand pixel is likely holding another pixel. We show that these can be learned from automatically generated pseudo-labels that use optical flow [21], epipolar geometry [15], and person masks [19]. Our network, named *MOVES*, learns a mapping to a per-pixel embedding; this embedding is then analyzed by grouping and association heads that are trained by cross-entropy to predict the pseudo-labels. While the pseudo-labels themselves

are poor and incomplete, we show that the learned classifiers are effective and that the embeddings are good enough to be analyzed by off-the-shelf, unspecialized algorithms like HDBSCAN [31]. Excitingly, even though our signal comes only when objects are picked up, our features generalize to objects that are not currently being interacted with.

We train and evaluate *MOVES* on challenging egocentric data, including EPIC-KITCHENS [6, 7] and EGO4D [13]. Our experiments show that once trained, *MOVES* features enable strong performance on a number of tasks related to hands and the objects they hold. First, using *MOVES* on the COHESIV [37] hand-object segmentation benchmark for EPIC-KITCHENS [6] improves by 31% relative (19.5 $\rightarrow$ 25.7) over the recent weakly-supervised COHESIV method [37] in object segmentation. Second, we show that distance in *MOVES* feature space is strongly predictive of two pixels being part of the same object, as well as a *Box2Seg* task where *MOVES* features are trivially analyzed to upgrade bounding-box annotations to segments. We show that *Box2Seg* shows strong performance on both objects that are currently being held as well as objects that *are not held* (unlike past work). In particular, compared to COHESIV, we show a strong gain on segmenting held objects (8.9 $\rightarrow$ 44.2 mIoU) as well as non-held objects (7.5 $\rightarrow$ 45.0 mIoU). Finally, we show that we can train an instance segmentation model [26] on the *Box2Seg* annotations and get good models for rough instance segmentation.

2. Related Work

Our work aims to learn to segment hands and objects that hands hold in new scenes from a single RGB image by observing videos at training time. Our work is distinguished from past work by three characteristics: *weak supervision*, since it uses flow and people masks rather than precise annotations; *associating hands with objects*, since it learns to identify what objects are currently held by hands; and *segmenting background objects*, since its learned embedding also works on objects that hands are not currently holding (i.e., background objects).

Hands and held-objects. There has been extensive work on hands, ranging from 2D detection and segmentation of hands [2, 9, 32, 36, 37] to 3D reconstruction of hands and objects [3, 16, 34]. Many of these works are supervised by human annotations. For instance, [9] provides tens of thousands of detailed annotations of hands and objects used on the EPIC-KITCHENS [6, 7] dataset. Our work differs from these supervised works by aiming to learn from a video signal at training time. This puts it as part of a line of work [12,37] that aims to use a little bit of information about humans to extract rich information from egocentric videos. Of these works, the most similar is COHESIV [37] which uses contrastive learning to separate hands, held-objects and background. Unlike COHESIV, our system also segments background objects using supervision gleaned while similar objects were held in training videos.

Motion coherence for object discovery. Although hands and held-objects make up the "what" of our signal, it is motion coherence that makes up the "how". Motion has been known as a signal for perceptual grouping since Gestalt psychologists first proposed ideas of common fate [43]. Ideas of common fate led to the usage of optical flow in tasks requiring motion segmentation [38]. We only use optical flow during training, but ignore it during testing, like other recognition approaches [33]. This puts our work as part of a long line of work on using flow to discover grouping [27, 46]. Our work is separated from much of this work by focusing on hands and providing not just *grouping* information but also an *association* mapping hands to hand-held objects. Regardless of the additional learned association model, our work is additionally separated from work in this space by: training on real egocentric data as opposed to synthetic 3rd person data such as [22]; and grouping static objects in addition to dynamic objects such as [24, 28]. Existing work does group static objects, but first learns these groupings dynamically [44]. Our work shares high-level goals with methods like EISEN [5] and the concurrent [23] but differ in a few ways: we assume knowledge of an agent and learn an association between the agent and objects; we test on egocentric data like EPIC-KITCHENS [8]; and we show good performance with the fusion of a simple discriminative objective with the right simple geometry-driven pseudolabels and large-scale egocentric data.

Unsupervised and weakly-supervised segmentation. There is a significant existing body of work learning segmentation in an unsupervised manner. Much of this existing work ends up performing classification implicitly, then leveraging the features to perform segmentation, as in DINO [4]. Works aside from object categorization include [20], which learns spatial and temporal co-occurence, as well as [41], where saliency is used as a class agnostic signal. Other work similarly focused on foreground segmentation as in [14], or adversarially predicted foreground motion as in [45]. Compared to most of this body of work, our work uses a complementary signal, namely motion, and also builds associations between hands and objects.

3. Method

Our system accepts a single RGB image and produces an embedding $\mathbf{E}$ that can be used to answer questions about: *grouping*, or whether two pixels belong to the same object; and *hand association*, whether one hand pixel is in contact with another object's pixel. We combine a per-pixel embedding network with two MLP heads that produce grouping and association probabilities from pairs of embeddings. These networks are trained to minimize a cross-entropy loss

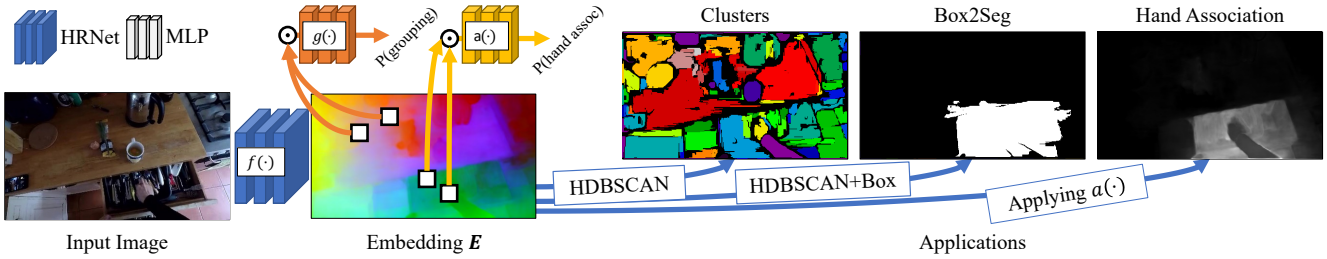

Figure 2. *MOVES* **Inference.** As input *MOVES* accepts an RGB image and produces a $H \times W \times F$ per-pixel feature embedding using a backbone HRNET [42] denoted $f(\cdot)$. Pairs of F-dimensional embeddings from this backbone can be passed to lightweight MLPs $g(\cdot)$ to assess grouping probability and $a(\cdot)$ to identify hand association, or if the pixels are a hand and an object the hand is holding. Once trained, the *MOVES* embeddings (here visualized with PCA to map the feature dimension to RGB) can be used for: *(Clusters)* directly applying HDBSCAN [31] to the embeddings produces a good oversegmentation; *(Box2Seg)*: Given a box, one can produce a more accurate segment; and *(Hand Association)* Applying a to a query point and every pixel produces hand-object association (here, to a drawer).

on pseudo-labels per task. We show that surprisingly simple pseudolabels that take into consideration epipolar geometry and people lead to highly effective training of our model. Once trained, we can directly use the embeddings and classifier heads to understand hands, hand-held objects, and objects that hands might hold later.

3.1. Architecture and Training

MOVES consists of a backbone embedding network and two heads that operate on embeddings from the network. As shown in Figure 2, the backbone f: $\mathbb{R}^{H\times W\times 3} \rightarrow \mathbb{R}^{H\times W\times F}$ converts an image to per-pixel F-dimensional feature embeddings. The grouping head g: $\mathbb{R}^{2F} \rightarrow [0,1]$ classifies whether two pixel embeddings go together as a binary classification; and the association head h: $\mathbb{R}^{2F} \rightarrow [0,1]$ classifies if two pixel embeddings represent a hand and a held object. The embedding backbone is a HRNET [42], without pre-training, and both classification heads are 3-layer MLPs.

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 $\text{CE}(y, \hat{y})$) applied to the classification head outputs, or:

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

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., $\mathbf{e}_{i,j} = [\mathbf{E}[i], \mathbf{E}[j]]$) and W is a per-image reweighting defined in §3.2 that indicates the quality of the pseudo-labels. We assume that the binary cross-entropy loss ignores any unknown i.e., 0 labels. We draw samples $\mathcal{S}$ randomly such that positives and negatives are equal and each foreground connected component is sampled proportionately.

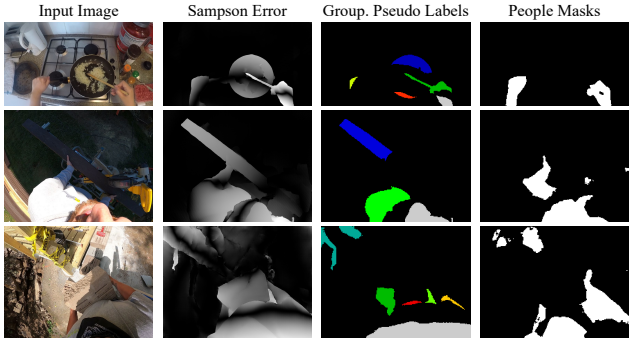

Figure 3. **Example Pseudolabels** For each image (top: EPIC-KITCHENS [6], middle: EGO4D [13], bottom: failure from EGO4D [13]), we show: the input image, the per-pixel Sampson error with respect to the fit fundamental matrix, the pseudolabels generated from connected components on thresholded Sampson error; and people masks from [19]. The pseudolabels are *partial* labels that are primarily unknown/unlabeled and only correct on-average. However, so long as failures do not have particular patterns, the network can treat them as noise.

3.2. Pseudolabels

To train this system, we need the ability to pseudo-label pixel relationships $\mathbf{G}_{i,j}$ and $\mathbf{A}_{i,j}$ in data. We propose an extraordinarily simple pseudo-label scheme that has high precision but perhaps low recall. While the pseudo-labels are grossly inadequate in any one image, using them to train a network on thousands of images leads to effective features. Our core assumptions are: that agents move objects and are in contact while doing so; that most visible motion is due to scene motion; and that objects are spatially contiguous. These assumptions are largely valid in egocentric data.

Basic Signal. Given a second image, offset ∼0.5s, we compute optical flow with [21, 40] and a forwards/backwards-consistency mask using a threshold of 10px. For image pair $\mathbf{I}, \mathbf{I}'$, we make optical flow maps $\mathbf{O}, \mathbf{O}' \in \mathbb{R}^{H\times W\times 2}$.

Cycle-consistent correspondences for $\mathbf{I}$ are those within a threshold ϵ of $\|\mathbf{O}_{x,y} - \mathbf{O'}_{x',y'}\|_2$, where $[x', y']^T = [x, y]^T + \mathbf{O}_{x,y}$. W from Equation 1 is the proportion of cycle consistent pixels for a given ϵ, where $\epsilon = 10$ pixels. We then fit a fundamental matrix [15] $\mathbf{F}$ on the consistent flow using RANSAC [11] and the 8-point algorithm [15].

Connected Components. After selecting an $\mathbf{F}$ to be the background motion model, we measure the sampson epipolar distance [30] for every point in $\mathbf{I}$, producing per-pixel Sampson error. Candidate positive objects are all pixels with Sampson error greater than τ, where τ is set per image, as $\tau = 0.5 * (max(\mathbf{S}) + min(\mathbf{S}))$. Pixels with error above threshold τ are considered foreground pixels, and the remaining pixels background. Connected components [10] is run on this foreground mask, producing Z distinct groups.

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.

Pseudo-label Accuracy. Our approach generates labels that are incomplete on any one training image but effective for training. Pixels whose flow has high Sampson error [30] (with respect to $\mathbf{F}$) are unlikely to be background and so encouraging the grouping of spatially contiguous foreground is almost always correct. Pixels with low Sampson error are *mainly* background with some pixels that are mistakenly put in the background due to motion alignment. However, which pixels get placed in the background depends on the camera pose of the second image used to compute flow; while this image is used to generate training signal, the network never sees it. Therefore, we hypothesize that the network treats these mistakes as incompressible noise.

Training Weight W. We set the per-image W in Eqn. 1 to the proportion of pixels with cycle-consistent optical flow, up-weighting more consistent and likely reliable images.

3.3. Using MOVES

Given a new image $\mathbf{I}$, the embedding $\mathbf{E} = f(\mathbf{I})$ produced by *MOVES* enables simple approaches to many applications in the hands and hand-held object literature.

Clustering for Objects. We find that the trained embeddings $\mathbf{E}$ can be directly clustered with HDBSCAN [31]. In fact, despite being trained only on held objects, we find that the network can do well on non-held objects. Using the embeddings rather then pairwise classification via g avoids a quadratic number of MLP evaluations.

Hand-Object Association. *MOVES* can take a point on a hand and identify *what goes with this pixel*, as studied in COHESIV [37]. Given a query at pixel i, we compute an association prediction $\mathbf{H} \in \mathbb{R}^{H \times W}$ where for each pixel j, we compute $\mathbf{H}_j = a(\mathbf{e}_{i,j})$ where $\mathbf{e}_{i,j} \in \mathbb{R}^{2F}$ is the concatenation of the ith and jth pixel of $\mathbf{E}$. The prediction is then improved via the clusters: for each cluster, we update its pixels with the average of in-cluster value of $\mathbf{H}$.

Box2Seg. The precise definition of an object is often a-priori unclear. Consider peering into a refrigerator: is the cap separate from a carton; is the carton separate from the fridge door; and is the fridge door separate from the fridge? If we knew the extent that was requested (e.g., the carton, not the cap), we could confidently segment the object.

Given a box to show rough extent, *MOVES* can convert a box to a segment. Inspired by the superpixel straddling cue of [1], we take the clusters and accept any that are cleanly within the box (<5% of its area outside the box). If there is no such cluster, we take the largest cluster inside the box.

3.4. Implementation Details

We train models on 8 GTX 2080 GPUs with batch size 16. *(Training)* We minimize Eqn. 1 with AdamW [29], with an initial LR of 10^{-4}. During training, we reduce the learning rate by a factor of 0.5 when validation loss plateaus after 5 mini-epochs of 2500 training samples. We halt training after 10 mini-epochs without a validation loss reduction. *(Feature dimension)* we set the feature dimension F of the embedding to be 128. *(Frame Size)* images are downsized to (576, 1024) in EpicKitchens and (648, 864) in Ego4D.

4. Experiments

We evaluate how well *MOVES* can understand hands and the objects that they hold via a series of experiments that assess three questions. First, in Section 4.1, we ask whether our features and classifiers can understand hands and the objects that they hold. We evaluate our method on the COHESIV [37] benchmark built on top of EPIC-KITCHENS [6, 8]. Second, in Section 4.2, we investigate how well our features group objects together. Here, we focus not only on hands and objects that are currently being held, but also on objects that appear in the background. We test our system on both EPIC-KITCHENS [6, 8] and Ego4D [13]. Finally, in Section 4.3, we use our system to upgrade all of the boxes in a dataset to segment and analyze how well a system trains on these.

4.1. Hand-Held Object Segmentation

We first evaluate on the hand and hand-held object benchmark built by COHESIV [37] on top of the EPIC-KITCHENS [6] dataset. In the COHESIV benchmark, each method is given a pixel corresponding to a hand and must produce a segment for that hand as well as a segment for

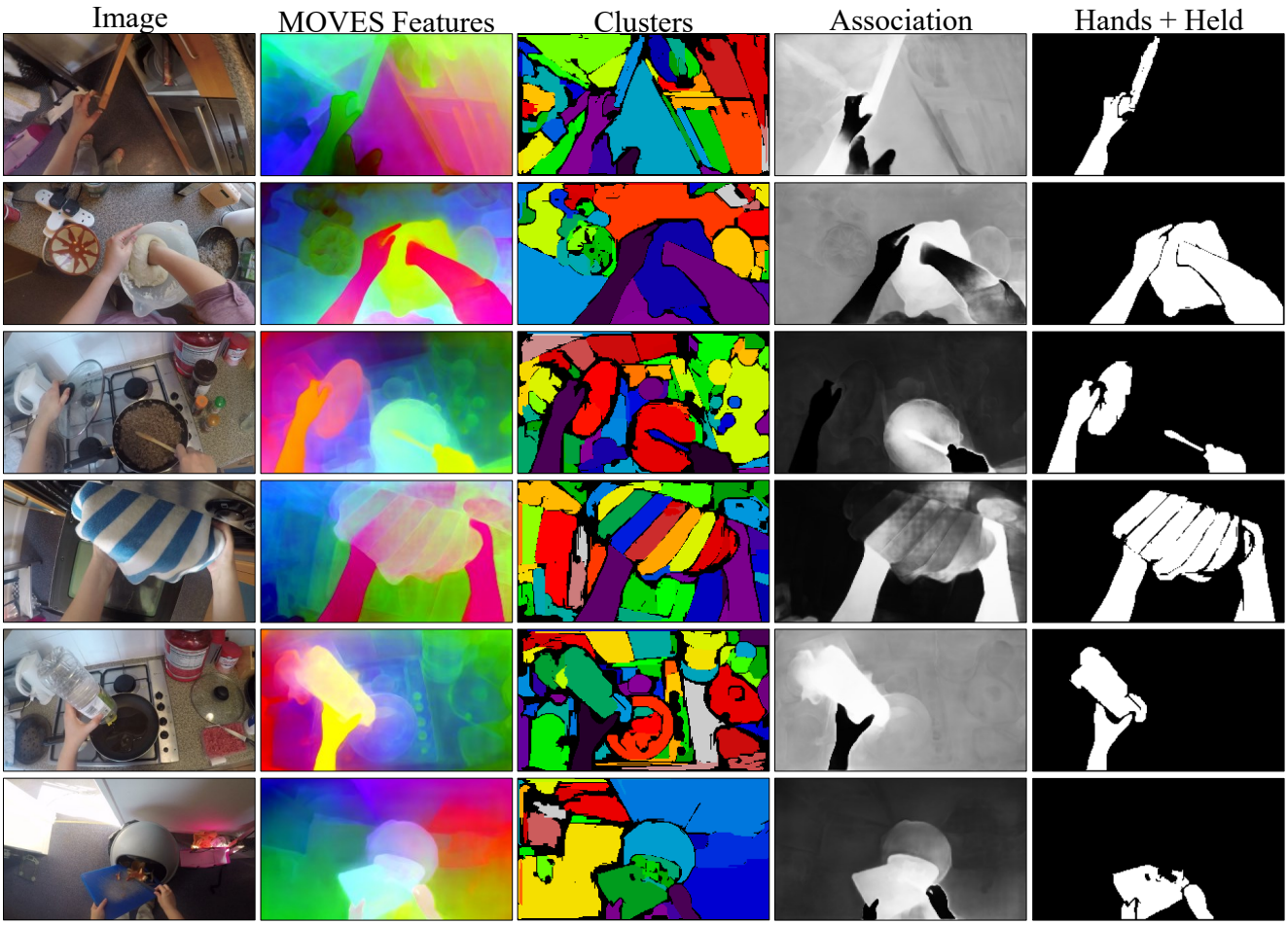

Figure 4. Results from *MOVES*, with examples from the EPICK VISOR validation set. **Key:** Each column shows a different input image. From left to right we show: (*Image*) the input image; (*MOVES Features*) a PCA projection of the feature space to RGB; (*Clusters*) The clusters found by HDBSCAN applied to the feature space with each cluster visualized with a random color; (*Association*) The prediction of the association head on the image on one of the hands in the image; (*Hands+Held*) A Mask of hands and hand-held Objects in the image. The association head usually does a good job of recognizing the objects that hands are holding. **Discussion:** (row 1) although the cabinet door is thin, *MOVES* recognizes the association between hand and door. (row 2) *MOVES* detects a large mixing bowl. (row 3) the transparent glass pan lid is recognized as an object by *MOVES* despite the stovetop below being visible through it. (row 4) the multi-colored hand towel is clustered as separate segments, however the association head helps segment most of the hand towel, showing the complementary nature of pairing an association head with clustering. (row 5) the transparent bottle is segmented nicely. (row 6) the cutting board is being cleared into the trashcan, but *MOVES* successfully identifies the board as being the held object.

the held object. Estimating the extent of the held object is usually unambiguous, but the hand is not, since there are a wide variety of definitions for the hand extent: VISOR [9] defines it as including the arm, while 100DOH [36] and COHESIV [37] define it as terminating at the wrist.

Dataset and Metrics. We evaluate on the COHESIV setup. This setup consists of pixel-labelings for the hands up to the wrist as well as the held object per hand. This produces a set of evaluation settings, each evaluated by the average pixel intersection over union (IoU): *(Object)* the object the hand is holding is positive; *(Hand)* the hand up to the wrist is positive; *(Pair)* both this particular hand and its held object are positives; as well as *(All)* where all hands and hand-held objects are positive.

The *MOVES* Solution. Given a pixel, we produce the held object using the Hand-Object Association inference in §3.3. We produce the hand by taking the embedding cluster from that the hand pixel falls into. This cluster agrees with the VISOR [9] definition of hand and includes the forearm; to align it with the COHESIV wrist definition, we clip it with the inferred bounding box from [36].

Baselines. We compare with COHESIV [37] as well as its baselines. Two low-level cue methods aim to test for alternate explanations for the results: Flow [40] is optical flow

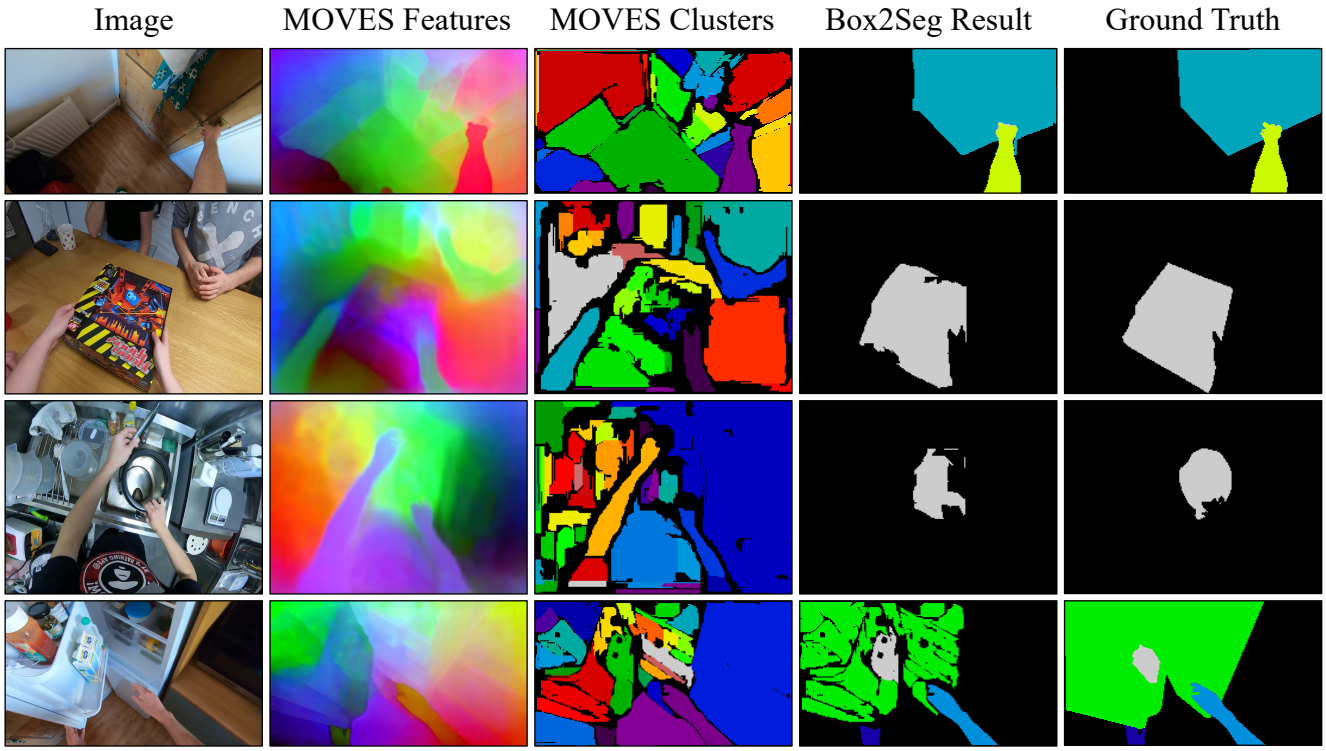

Figure 5. **Box2Seg Output.** (*Image*) input images from EPICK [6] (rows 1 and 4) and Ego4D [13] (rows 2 and 3); (*MOVES Features*) a PCA visualization of the learned *MOVES* embedding space; (*Clusters*) The clusters found by HDBSCAN; (*Box2Seg Result*) our *Box2Seg* result which agglomerates clusters for segmentations at different scales; (*Ground Truth*) Annotations from VISOR [9] or FG [47]. For each annotated object, a mask is shown in the corresponding color. Our clusters are often slight oversegmentations of objects. However, knowing the spatial extent of the object helps resolve ambiguity about parts of objects vs objects.

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

given a future frame. This controls for the possibility that systems' predictions just amount to predicting pixels that are likely to move. Saliency [48] is a salient region detection system that controls for whether the system is just predicting objects that visually stand out. The supervised system [36] is an object detection system that is trained on over 100K labeled images with boxes. Its outputs are strong approximations for boxy objects and poor for thin objects.

Qualitative Results. We show some examples of the learned groupings and associations in Figure 4. Our learned association head produces a good estimate of the objects that hands are currently holding and our clusters tend to align well with object boundaries. This is true for the objects people are holding, as is evaluated in this experiment and as was the case with COHESIV too. However, *MOVES* also produces good clusters for the background objects as well. We more thoroughly evaluate these background objects in the subsequent Section 4.2.

Quantitative Results. We show quantitative results in Table 1. *MOVES* outperforms COHESIV [37] in all categories, and the supervised method [36] on hands. As an added bonus, our inference for objects is substantially simpler than COHESIV: we run a MLP at each location, while [37] reports fitting a model to pseudo-labeled embeddings on the training set, and merging two independent predictions. This puts the method within 0.2 mIoU of the supervised [36] on the Pair metric. Notably, this benchmark only tests objects when they are being held; we find substantially higher performance on all objects in the next section.

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

4.2. Object Segmentation

We next turn to evaluating how well we can segment objects, this time including boxes that are *not* currently being held by a hand. Note that our training signal exclusively comes from objects as they are held by hands. However, by learning to group pixels together, our system learns grouping cues that generalize to non-held objects.

Datasets. We evaluate on two datasets to demonstrate the effectiveness of the system.

VISOR [9]. The first dataset is the VISOR [9] benchmark for EPIC-KITCHENS [6, 8], which has precisely annotated segments for *active* objects that are part of ongoing long-term activities but are not necessarily currently in contact with hands. We use them as a source of bounding boxes (used as input) and segments (used as ground-truth) for objects. We divide these into *Held* objects that are held by a hand and *Non-Held* objects that are not held by a hand.

Fine-Grained [47]. Our second dataset is the Fine Grained [47] annotations on top of the EGO-4D [13] dataset. These *do not* have inactive objects. To obtain features, we train our method using identical settings and hyperparameters on Ego4D [13].

Settings and Metrics. We use two settings, each with their own metric. First, we apply *Box2Seg* of §3.3 (essentially: cluster with HDBSCAN [31], take clusters that lie entirely in the box). This produces a hard prediction about the segmentation. We then evaluate the predicted segmentation using the mean intersection over union (mIoU) to quantify the quality of the obtained masks.

Second, to directly evaluate how well feature spaces predict object segmentation, we test whether distances between per-pixel embeddings predicts whether being in the same object. We are given a pixel i on an object and pixel i' in the image and compute the per-pixel feature embeddings $\mathbf{E}_i$ and $\mathbf{E}_{i'}$. We then compute whether similarities between pixels (i.e., the negative of the distance or $-||\mathbf{E}_i - \mathbf{E}_{i'}||$) predict that pixel i' is on the same object as i and evaluate performance with the area under the receiver operating characteristic (AUROC). The AUROC represents the chance that a positive sample has a higher score than a negative sample. To define positives and negatives, we use the ground-truth mask. We compute the AUROC per-object, sampling five on-object pixels to be compared with all other pixels. We then report the average AUROC.

Table 3. **Box2Seg Feature Distance Evaluation.** To directly test feature spaces, we also compute an AUROC-based evaluation. We report the discriminative power of feature space distances in predicting that two pixels are part of the same object. *MOVES* again shows strong performance both by itself as well as relative to baselines.

	VISOR [9]			FG [47]	
	Hand	Held	Non	Hand	Held
MOVES	**99.5**	**95.2**	**95.0**	**97.1**	**94.7**
COHESIV [37]	81.3	75.9	77.3	72.4	78.1
Pretrained [42]	71.9	62.9	62.3	71.5	63.9
RGB	81.5	78.4	78.6	75.8	73.3

Figure 6. **MOVES Clusters vs COHESIV Clusters.** For the input image on the left, we show clusters found by *MOVES* and clusters found by COHESIV [37]. While COHESIV produces reasonable clusters on the hands and objects near the hands, it collapses the rest of the image's feature space into a single cluster. Similar behavior can be seen in Fig. 3 of [37].

The *MOVES* Solution. We use the learned embedding space $\mathbf{E} = f(\mathbf{I})$. For hard predictions, we run the *Box2Seg* method from §3.3. For distances, we use feature space distances directly. We show example outputs of Box2Seg in Fig. 5, showing our clusters recovered by HDBSCAN [31] as well as linear projections of feature space on which we measure distances.

Baselines. We apply *Box2Seg* to three other different feature representations. The first is (**COHESIV** [37]), which is also learned via a weakly-supervised objective. Among a number of differences, COHESIV radically differs in how it treats non-held pixels: non-held background pixels are pulled together to organize the latent space. This compares our method against a comparable recent method. The second is a pretrained network (**Pretrained**) that consists of features from an ILSVRC-pretrained [35] ResNet [18] backbone. This baseline compares our method with off-the-shelf features trained on ImageNet [35]. These often serve as a strong baseline on recognition tasks. The last **(RGB)** is raw RGB features, which tests a simple low-level cue of distance in RGB color space. A fraction of the test objects

can be segmented out by simple color cues, but in realistic egocentric data most cannot be.

Results. We show quantitative results in Table 2 for mIoU and Table 3 for AUROC. Our approach, *MOVES*, substantially outperforms the baselines. Our finding that off-the-shelf ImageNet features do not work is in line with prior work [12], which showed that often these features are not effective for tasks like clustering on egocentric data. COHESIV [37] does on-par with RGB features in this task for all objects *except* hands. At first glance, this seems surprising because COHESIV also is trained and therefore ought to nonlinearly transform the feature space into something more amenable to recovering objects. We illustrate why in Figure 6. The COHESIV objective pulls all background pixels towards the same label in order to stabilize its training. This has the downside of collapsing the latent space so that background objects have similar embeddings. Hands clearly stick out, but the entire background is grouped.

One hypothesis for less good *Box2Seg* results is that there is a misalignment between the *Box2Seg* method and the baselines' features. We test this hypothesis by evaluating the AUROC on pairs of distances. This evaluation tests the discriminative power of the feature space distances. Our approach performs well on an absolute basis (with AUROCs all above 95%) and substantially outperforms the baselines.

Ablations. We evaluate how pseudolabel design and optical flow impact performance. We use GMA-Flow [21], and find that using PWC-Net Optical flow [39] performs similarly, with Hand, Held, and Non feature distance results (evaluated equivalently to Table 3) of 98.2, 94.6, and 94.2. We also evaluate alternative pseudolabels, made by running connected components on pixels with norm optical flow greater than per-image mean, and find similar Hand, Held, and Non results of 98.9, 94.4, and 94.5.

Additional Comparisons. We compare *MOVES* against DINO [4] and EISEN [5]. DINO achieves a worse AUROC score (evaluated equivalently to Table 3) of 93.1, 91.1, and 89.3 on Hand, Held, and Non distance evaluation results for VISOR. EISEN's playroom is a dataset with no people, and as such we train a version of *MOVES* using only the grouping loss. We find that *MOVES* achieves a comparable mIoU (71.8 compared with EISEN's 73.0) out-of-the-box, with default settings. This comparable performance suggests that EISEN's affinity graphs, graph propagation, and competition are not needed. Instead, a straightforward grouping signal is sufficient to segment objects.

4.3. Using MOVES To Train Instance Segmentation

In §4.2, we evaluated using *MOVES* to convert boxes to segments. As a proof of concept, we evaluate a potential application of this: using *MOVES* to accelerate the annotation of datasets. Since boxes are much cheaper to annotate than segments, one could potentially label boxes instead and then automatically annotate interacted-with objects.

Table 4. As a proof of concept, we use *MOVES* predicted masks to re-train an instance segmentation system for the VISOR [9] HOS Challenge. While our system falls short of a system trained on painstakingly annotated images, it performs acceptably.

	Mask AP@50		
	Hand	Held	Active
MOVES Box2Seg	81.7	25.5	22.3
Supervised Upper Bound	**96.8**	**49.8**	**42.2**

Dataset and Metrics. We evaluate on the HOS benchmark of the VISOR [9] benchmark suite. This consists of precisely annotated segments of hands and hand-held objects. There are two tasks: *Active* objects, or any object that is currently being used in the activity (but which may not currently be in contact); as well as Hands and Hand-Held Objects. We report performance for Mask AP evaluated at IoU of 50%. COCO AP evaluates over multiple IoU thresholds with much of the performance evaluated at stringent IoU requirements (e.g., 95% IoU) that make sense for learning from precisely annotated datasets, but less sense for automatically labeled data.

The *MOVES* Solution. We run *Box2Seg* on the bounding box annotations of the VISOR dataset and train using the same PointRend [26] network and box annotations. The only annotation change is the segments.

Results. We compare with training with supervised annotations. Without segmentation labels, *MOVES* recovers half the performance of painstakingly annotated data on objects, and most of the performance on hands. Our approach to identifying segments for hands is not specialized for hands, but is just the *Box2Seg* method. If we intersect our *Box2Seg* hands with the people masks of [19], we improve AP50 by 10 points to 91.7.

5. Conclusion

Picking up an object provides a powerful signal about grouping that instantly disambiguates which pixels go together. This principal has been known since the Gestaltists [43], but making it work in practice has proved elusive in computer vision. Past works in the hand-and-hand-held object space [9, 36, 37] have difficulty segmenting objects in the background: when not in use, objects fade into a monolithic inactive category. Our paper shows that one can glean supervision from small amounts of information in a video, such as flow [40], epipolar geometry [15], and per-pixel identification of people [19]. By learning from a large dataset like EPIC-KITCHENS [6, 8], one can build surprisingly effective features with extremely simple and direct discriminative training.

References

[1] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2189–2202, 2012. 4

[2] Sven Bambach, Stefan Lee, David J Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *ICCV*, 2015. 2

[3] Zhe Cao, Ilija Radosavovic, Angjoo Kanazawa, and Jitendra Malik. Reconstructing hand-object interactions in the wild. In *ICCV*, 2021. 2

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 2, 8

[5] Honglin Chen, Rahul Venkatesh, Yoni Friedman, Jiajun Wu, Joshua B Tenenbaum, Daniel LK Yamins, and Daniel M Bear. Unsupervised segmentation in real-world images via spelke object inference. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIX*, pages 719–735. Springer, 2022. 2, 8

[6] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *ECCV*, 2018. 2, 3, 4, 6, 7, 8

[7] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 2, 6

[8] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision. *CoRR*, abs/2006.13256, 2020. 2, 4, 7, 8

[9] Ahmad Darkhalil, Dandan Shan, Bin Zhu, Jian Ma, Amlan Kar, Richard Ely Locke Higgins, Sanja Fidler, David Fouhey, and Dima Damen. Epic-kitchens visor benchmark: Video segmentations and object relations. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*. 1, 2, 5, 6, 7, 8

[10] Luigi Di Stefano and Andrea Bulgarelli. A simple and efficient connected components labeling algorithm. In *Proceedings 10th international conference on image analysis and processing*, pages 322–327. IEEE, 1999. 4

[11] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981. 4

[12] Mohit Goyal, Sahil Modi, Rishabh Goyal, and Saurabh Gupta. Human hands as probes for interactive object understanding. In *CVPR*, 2022. 1, 2, 8

[13] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *CVPR*, 2022. 1, 2, 3, 4, 6, 7

[14] Emanuela Haller and Marius Leordeanu. Unsupervised object segmentation in video by efficient selection of highly probable positive features. In *ICCV*, 2017. 2

[15] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 1, 4, 8

[16] Yana Hasson, Bugra Tekin, Federica Bogo, Ivan Laptev, Marc Pollefeys, and Cordelia Schmid. Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In *CVPR*, 2020. 2

[17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 1

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 7

[19] Vladimir Iglovikov and Alexey Shvets. Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation. *arXiv preprint arXiv:1801.05746*, 2018. 1, 3, 4, 8

[20] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson. Learning visual groups from co-occurrences in space and time. *arXiv preprint arXiv:1511.06811*, 2015. 2

[21] Shihao Jiang, Dylan Campbell, Yao Lu, Hongdong Li, and Richard Hartley. Learning to estimate hidden motions with global motion aggregation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9772–9781, 2021. 1, 3, 8

[22] Rishabh Kabra, Daniel Zoran, Goker Erdogan, Loic Matthey, Antonia Creswell, Matt Botvinick, Alexander Lerchner, and Chris Burgess. Simone: View-invariant, temporally-abstracted object representations via unsupervised video decomposition. *NeurIPS*, 2021. 2

[23] Laurynas Karazija, Subhabrata Choudhury, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Unsupervised multi-object segmentation by predicting probable motion patterns. *arXiv preprint arXiv:2210.12148*, 2022. 2

[24] Thomas Kipf, Gamaleldin F Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. *arXiv preprint arXiv:2111.12594*, 2021. 2

[25] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, 2019. 1

[26] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *CVPR*, 2020. 2, 8

[27] Yin Li, Manohar Paluri, James M Rehg, and Piotr Dollár. Unsupervised learning of edges. In *CVPR*, 2016. 2

[28] Runtao Liu, Zhirong Wu, Stella Yu, and Stephen Lin. The emergence of objectness: Learning zero-shot segmentation from videos. *NeurIPS*, 2021. 2

[29] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 4

[30] Quang-Tuan Luong, Rachid Deriche, Olivier Faugeras, and Theodore Papadopoulo. *On determining the fundamental matrix: Analysis of different methods and experimental results*. PhD thesis, Inria, 1993. 4

[31] Leland McInnes, John Healy, and Steve Astels. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, 2(11):205, 2017. 2, 3, 4, 7

[32] Supreeth Narasimhaswamy, Trung Nguyen, and Minh Hoai. Detecting hands and recognizing physical contact in the wild. In *NeurIPS*, 2020. 2

[33] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *CVPR*, 2017. 2

[34] J. Romero, D. Tzionas, and M. J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), 2017. 2

[35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 7

[36] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F Fouhey. Understanding human hands in contact at internet scale. In *CVPR*, 2020. 1, 2, 5, 6, 8

[37] Dandan Shan, Richard Higgins, and David Fouhey. Cohesiv: Contrastive object and hand embedding segmentation in video. *NeurIPS*, 2021. 1, 2, 4, 5, 6, 7, 8

[38] Jianbo Shi and Jitendra Malik. Motion segmentation and tracking using normalized cuts. In *ICCV)*, 1998. 2

[39] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 8

[40] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020. 3, 5, 6, 8

[41] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. *ICCV*, 2021. 2

[42] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020. 3, 7

[43] Max Wertheimer. Laws of organization in perceptual forms. 1938. 1, 2, 8

[44] Yanchao Yang, Brian Lai, and Stefano Soatto. Dystab: Unsupervised object segmentation via dynamic-static bootstrapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2826–2836, 2021. 2

[45] Yanchao Yang, Antonio Loquercio, Davide Scaramuzza, and Stefano Soatto. Unsupervised moving object detection via contextual information separation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 879–888, 2019. 2

[46] Zhenheng Yang, Peng Wang, Yang Wang, Wei Xu, and Ram Nevatia. Lego: Learning edge with geometry all at once by watching videos. In *CVPR*, pages 225–234, 2018. 2

[47] Lingzhi Zhang, Shenghao Zhou, Simon Stent, and Jianbo Shi. Fine-grained egocentric hand-object segmentation: Dataset, model, and applications. In *ECCV*, 2022. 1, 6, 7

[48] Ting Zhao and Xiangqian Wu. Pyramid feature attention network for saliency detection. In *CVPR*, 2019. 6