Discovering Objects that Can Move

Zhipeng Bao[*,†,1] Pavel Tokmakov[*,2] Allan Jabri[3]
Yu-Xiong Wang[4] Adrien Gaidon[2] Martial Hebert[1]
[1]CMU [2]Toyota Research Institute [3] UC Berkeley [4] UIUC

Abstract

This paper studies the problem of object discovery – separating objects from the background without manual labels. Existing approaches utilize appearance cues, such as color, texture, and location, to group pixels into object-like regions. However, by relying on appearance alone, these methods fail to separate objects from the background in cluttered scenes. This is a fundamental limitation since the definition of an object is inherently ambiguous and context-dependent. To resolve this ambiguity, we choose to focus on dynamic objects – entities that can move independently in the world. We then scale the recent auto-encoder based frameworks for unsupervised object discovery from toy synthetic images to complex real-world scenes. To this end, we simplify their architecture, and augment the resulting model with a weak learning signal from general motion segmentation algorithms. Our experiments demonstrate that, despite only capturing a small subset of the objects that move, this signal is enough to generalize to segment both moving and static instances of dynamic objects. We show that our model scales to a newly collected, photo-realistic synthetic dataset with street driving scenarios. Additionally, we leverage ground truth segmentation and flow annotations in this dataset for thorough ablation and evaluation. Finally, our experiments on the real-world KITTI benchmark demonstrate that the proposed approach outperforms both heuristic- and learning-based methods by capitalizing on motion cues.

1. Introduction

Objects are the key building blocks of perception [31, 50]. We understand the world not in terms of pixels, surfaces, or entire scenes, but rather in terms of individual objects and their combinations. Object-centric representation makes tractable higher-level cognitive abilities such as casual reasoning, planning, etc., and is crucial for generalization and adaptation [5, 60]. In computer vision, progress has been achieved in object recognition recently [9, 24, 46], but these approaches rely on large amounts of expensive manual labels, and only cover a fixed vocabulary of object categories. Discovering objects and their extent in data – in a manner that generalizes across domains – remains an open problem.

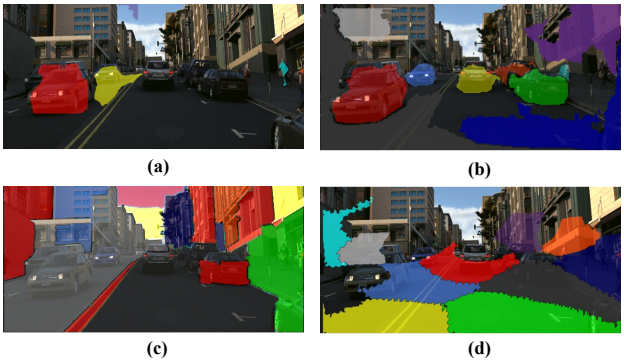

Figure 1. A sample from the TRI-PD dataset with: (a) motion segmentation from [14], top-10 segments produced by (b) our approach, (c) heuristic-based MCG [3], and (d) learning-based SlotAttention [38]. Our method uses noisy, sparse motion segmentation to learn to separate both moving and static instances of dynamic objects from the background, whereas others cannot resolve the object definition ambiguity based on appearance alone.

What makes this task especially challenging is that the notion of an object is inherently ambiguous and context-dependent. Consider a car in Figure 1: its left door and the handle on that door can be treated as individual objects, or parts of the whole. It is thus not surprising that existing approaches that attempt to automatically separate objects from the background based on appearance struggle beyond controlled scenarios. In particular, classical methods that use graph-based inference tend to over- or under-segment the objects [3, 18] (Figure 1, bottom left). More recent learning-based methods model object discovery with structured generative networks, often leveraging iterative inference in the bottleneck of an auto-encoder [8, 16, 22, 37, 38]. While promising results have been demonstrated, these approaches are typically restricted to toy images with colored geometric shapes on a plain background, and completely fail on realistic scenes (Figure 1, bottom right).

We posit that while the ambiguity of the object definition

[*]Equal contribution
[†]Work done during an internship at TRI

is not resolvable in the static image world without direct supervision, it has a natural resolution in the dynamic world of videos. Concretely, we choose to focus on *dynamic* objects, which we define as entities that are capable of moving independently in the world. Independent object motion is a strong grouping cue, which has been shown to drive object learning in animal perception [13,49]. In computer vision, there exists a long line of works on motion segmentation that automatically separate moving objects from the background based on optical flow [7, 14, 33, 41, 41, 61]. These methods have found numerous applications in unsupervised [2, 43] and weakly-supervised machine learning algorithms [27, 44, 56].

In this work, we show how motion segmentation can be bootstrapped to group instances even when they are static. We build our approach on top of the framework for unsupervised object discovery proposed by Locatello et al. [38], and show how to scale it from toy images to realistic videos. We extend the architecture to videos of arbitrary length by introducing a spatio-temporal memory module [4], and simplify the grouping mechanism to scale the model to realistic scenes with large resolution and dozens of objects. We then demonstrate the importance of inductive biases based on independent object motion on the emergent representation and the extent to which it captures objects. In particular, we show how motion segments (Figure 1, top left) can guide the attention operation to discover static objects. Crucially, we show that motion segmentation of varying quality – even when sparse and noisy – can be sufficient to bias the model towards segmenting *both moving and static instances* (Figure 1, top right). Our approach only requires videos for training, and can segment objects in static images at inference time.

To go beyond the toy data used in [38], while still being able to thoroughly analyze the various aspects of the method, we utilize a new, photo-realistic, synthetic dataset collected using the ParallelDomain platform [1] (TRI-PD). It consists of hundreds of videos, with crowded, street driving scenes, and comes with a full set of ground truth annotations, including object segmentations, 3D coordinates, and optical flow, allowing us to ablate the importance of the quality of the motion segmentation to the method's performance. Finally, we demonstrate that the resulting method generalizes to real videos on the challenging KITTI dataset [19], and compare it to existing heuristic- and learning-based approaches. Our code, models, and synthetic data are made available at https://github.com/zpbao/Discovery_Obj_Move/.

2. Related work

In this work we study the problem of *object discovery* in realistic videos capitalizing on *motion segmentation* as a *learning signal for bottom-up grouping*. Below, we review the most relevant works in each of these areas.

Object discovery is the problem of separating objects from the background without manual labels. Traditional computer vision approaches treated it as perceptual grouping [36] – the idea that low- and mid-level regularities in the data such as color, orientation, and texture allow for approximately parsing a scene into object-like regions. Notable approaches include [18], which uses graph-based inference to identify region boundaries, and [3] which first extracts regions on multiple scales with a normalized cut algorithm, and then groups them into object candidates. However, being purely appearance-based, these methods are not well equipped to resolve the inherent ambiguity of the object definition.

This problem has received renewed attention recently with the introduction of learning-based methods for object discovery [8, 16, 17, 22, 23, 29, 37, 38, 59, 64]. A common approach is to use iterative inference to bind a set of variables to objects in an image [16, 22, 38], usually with a variational auto-encoder [35, 47]. A more efficient variant is proposed by Locatello et al. [38] in their SlotAttention framework. Concretely, they perform a single step of image encoding with a CNN (convolutional neural network) followed by an iterative attention operation, which is used to bind a set of variables, called slots, to image locations. The slots are then decoded individually and combined to reconstruct the image.

Many of the approaches above are capable of discovering objects in toy, synthetic scenes, but as we demonstrate in Section 4.5, they fail in more realistic environments, where appearance alone is not sufficient to separate the objects from the background. In this work, we extend SlotAttention to realistic videos by modifying the architecture of the model to allow it to scale to large scenes with dozens of objects, and incorporating inductive biases in the form of motion segmentation. Crucially, our method only uses motion as a sparse learning signal and the trained model is able to segment both moving and static instances.

Finally, several works have recently explored integrating inductive biases in the form of 3D geometry constraints [11, 15, 26, 51]. However, these methods remain limited to toy, synthetic environments. In contrast, our method uses independent object motion as a learning signal, allowing it to generalize to real-world scenes. Geometric priors are orthogonal to our approach and combining different forms of inductive biases is a promising direction for future work.

Motion segmentation is concerned with separating objects from the background using optical flow [28, 53, 55]. Early approaches [7, 33, 41, 41] tracked individual pixels with the flow and then clustered the resulting trajectories inspired by the common fate principle [36]. While these methods have shown promising results on motion segmentation benchmarks, they do not generalize well in the wild due to their heuristic-based nature. More recently, several learning-based methods have been proposed [14, 61]. In particular, Dave et al. re-purpose a state-of-the-art object detection architecture [24] to detect and segment moving objects in an optical

flow field. The model is trained on a toy, synthetic FlyingThings3D dataset [39], but can generalize to real videos due to appearance abstraction provided by the flow. We use this method in our work due to its high performance and simplicity combined with minimal supervision requirements. Note that since our method requires instance-level moving object masks, binary motion segmentation techniques [42, 57, 62] are not applicable in our scenario.

Learning from motion is a paradigm inspired by evidence from cognitive science research, where independent object motion is a crucial cue for the development of the human visual system [49]. In computer vision, it has been adopted for weakly-supervised object detection [44] and semantic segmentation [27, 56], as well as for unsupervised representation learning [2, 43]. However, none of these works address the problem of object discovery from unlabeled videos. Yang et al. [63] use binary motion segmentation to train saliency models, but do not segment individual objects in complex scenes. Very recently, Tangemann et al. [54] have proposed to use motion segmentation to build compositional, generative scene models. However, their approach employs motion segmentation as a pre-processing step during training and is not capable of object discovery at inference time.

3. Method

In this section, we first introduce the SlotAttention framework for unsupervised object discovery, which serves as a basis for our approach, in Section 3.1. We then describe how we scale this architecture to real-world videos with dozens of objects in Section 3.2, and present our approach to incorporating independent motion priors in Section 3.3.

3.1. Background

Following prior work [8, 22], SlotAttention [38] models object discovery as inference in an auto-encoder framework. Concretely, given an image $I \in \mathbb{R}^{H \times W \times 3}$, it is first passed through an encoder CNN to obtain a hidden representation $H = f_{enc}(I) \in \mathbb{R}^{H' \times W' \times D_{inp}}$. It is then processed by an attention module, which we describe below, to map H to a set of feature vectors of a fixed length K called slots $S \in \mathbb{R}^{K \times D_{slot}}$. Each slot $S_i \in S$ is broadcasted onto a 2D grid, and decoded individually with a decoder CNN $O_i = f_{dec}(S_i) \in \mathbb{R}^{H \times W \times 4}$, where the 4th dimension of the output represents the alpha mask A_i. Denoting the first 3 channels of O_i with $I_i^{'}$, the complete image reconstruction is obtained via $I' = \sum_i A_i * I_i^{'}$ and is used to supervise the model with an MSE (mean squared error) loss.

The attention module is the key component of the approach. It uses an iterative attention mechanism, similar to the one used in Transformer [58], to map from the input H to the slots S. In particular, the attention weights are computed with a dot product between the input features and slot states $W = \frac{1}{\sqrt{D}} k(H) \cdot q(S) \in \mathbb{R}^{N \times K}$, where k and q are learnable linear transformations and $N = H' \times W'$. These attention weights are then used to compute the update values via $U = W^T v(H) \in \mathbb{R}^{K \times D}$, where W are the normalized attention weights, and v is another linear transformation. A key difference to the classical Transformer architecture is that the slots are initialized at random, and the inference is iterative. In particular, at every step l the slots are updated via $S_l = \texttt{update}(S_{l-1}, U_l)$, where the update function is implemented as a GRU (gated recurrent unit) [12].

The intuition behind this approach is that the slots serve as a representational bottleneck and individual decoding of the slots results in them binding to spatially coherent regions, such as objects. Next, we describe how we modify the SlotAttention framework to scale it to real-world videos.

3.2. A framework for object discovery in videos

Our model, shown in Figure 2, takes a sequence of video frames $\{I^1, I^2, ..., I^T\}$ as input. Following [38], each frame is then processed by an encoder CNN, shown in yellow, to obtain an individual frame representation $H^t = f_{enc}(I^t)$. These individual representations are aggregated by a ConvGRU spatio-temporal memory module [4] to obtain video encoding via $H^{'t} = \texttt{ConvGRU}(R^{t-1}, H^t)$, where $R^{t-1} \in \mathbb{R}^{H' \times W' \times D_{inp}}$ is the recurrent memory state.

Next, we proceed to map the video representation $H^{'t}$ to the set of slots S^t. It is easy to see, however, that the recurrent slot assignment strategy proposed in [38] does not scale well to sequential inputs. Indeed, given a sequence of length T and L inference steps for each frame, the overall number of attention operations required to process the sequence is $T \times L$. Such a nested recurrence is both computationally inefficient, and can exacerbate the vanishing gradient problem. To address this issue, as shown in the blue block in Figure 2, we only perform a single attention operation to directly compute the slot state $S^t = W^{t^T} v(H^{'t})$, where the attention matrix W^t is computed using the slot state in the previous frame S^{t-1}. For the first frame we use a learnable initial state S^0.

It is worth noting that the authors of [38] suggest that iterative inference on randomly initialized slots is crucial for the model to be able to generalize to a different number of objects at test time. However, we have found that simply increasing the number of slots to the maximum expected number of objects is sufficient to generalize to scenes of varying complexity. In that regard, our approach is similar to DETR [9], which also uses transformer query vectors as learnable object proposals that are capable of parsing both densely and sparsely populated scenes, but is trained in a fully supervised way.

Finally, the resulting slot states S^t need to be processed with the decoder CNN, shown in green in Figure 2, to obtain the frame reconstruction. However, the individual slot

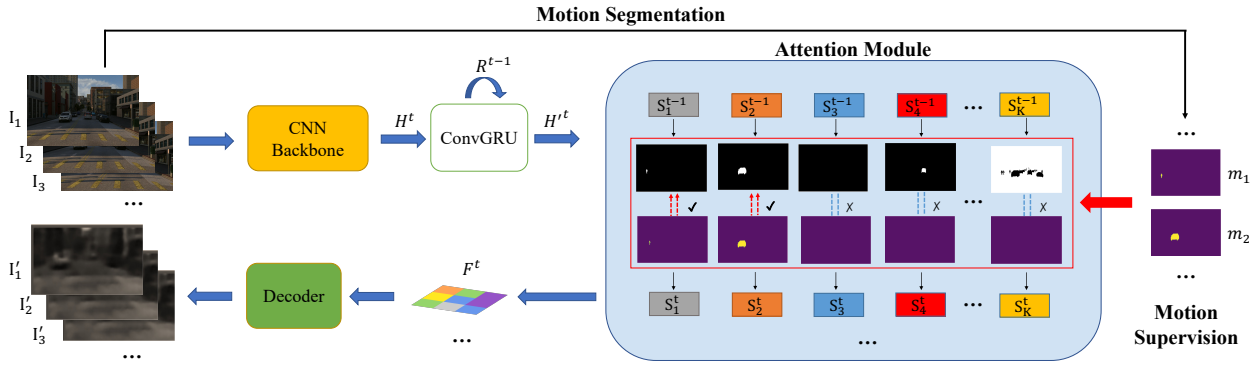

Figure 2. Our method takes a sequence of frames as input and processes them individually with a backbone network (shown in yellow), and a ConvGRU recurrent memory module. The resulting feature maps $H^{'t}$ are passed to the attention module (shown in blue) which binds them to a fixed set of slot variables via an attention operation. We additionally use automatically estimated motion segmentation to guide the attention operation for a subset of the slots. Finally, the slot states are combined in a single feature map F^t and decoded to reconstruct the frame. The reconstruction objective enforces generalization from moving to static instances.

decoding approach from [38] does not scale well with the number of slots. Indeed, a full image reconstruction needs to be computed for each slot which quickly becomes prohibitively expensive in terms of memory, especially for large resolution frames. Instead, we propose to invert the order of slot decoding and slot recombination steps. In particular, we first broadcast each individual slot feature $S_i^t \in \mathbb{R}^D_{slot}$ to a feature map $F_i^t \in \mathbb{R}^{H' \times W' \times D_{slot}}$ and use the attention mask $W^t_{:,i}$ of the slot as an alpha mask A_i^t. We then construct a single output feature map $F^t = \sum_i A_i^t * F_i^t$, shown with a checkerboard pattern in Figure 2, and decode it via $I^{'t} = f_{dec}(F_t) \in \mathbb{R}^{H \times W \times 3}$.

As we demonstrate in Section 4.3, the proposed single shot decoding strategy reduces the strength of the spatial cohesion prior to the original SlotAttention architecture, decreasing its object discovery capabilities. However, we also demonstrate that this prior does not generalize beyond toy, synthetic scenes. Instead, in the next section we describe our approach to incorporating an independent motion prior, which provides a stronger learning signal and works well with a single shot decoding strategy.

3.3. Incorporating independent motion priors

Our method assumes that a set of sparse, instance-level motion segmentation masks $\mathcal{M} = \{M^1, M^2, ..., M^T\}$ is provided with every video, with $M^t = \{m_1, m_2, ..., m_{C^t}\}$, where C^t is the number of moving objects that were successfully segmented in frame t, and $m_j \in \{0,1\}^{H' \times W'}$ is a binary mask (downsampled to match the spatial dimension of the feature maps). Note that for every frame it is possible that $M^t = \emptyset$. This reflects the realistic assumption that a variable number of objects can be moving in any given frame and that in some frames all the objects can be static.

We propose to use these motion segmentation masks to directly supervise the slot attention maps $W^t \in \mathbb{R}^{N \times K}$. We thus need to map a variable number of motion segmentations C^t to a fixed number of slots K in every frame. Following prior work on set-based supervision [9, 52], we first find an optimal bipartite matching between predicted and motion masks, and then optimize an object-specific segmentation loss. Specifically, we consider M^t also as a set of length K padded with $\emptyset$ (no object). To find a bipartite matching between these two sets we search for a permutation of K elements with the lowest cost:

$$\hat{\sigma} = \arg\min_{\sigma} \sum_{i=1}^{K} \mathcal{L}_{seg}(m_i, W^t_{:,\sigma(i)}), \tag{1}$$

where $\mathcal{L}_{seg}(m_i, W^t_{:,\sigma(i)})$ is the segmentation loss between the motion mask m_i and the attention map of the slot with index $\sigma(i)$. In practice, we efficiently approximate the optimal assignment with a greedy matching algorithm.

Once the assignment $\hat{\sigma}$ has been computed, the final motion supervision objective is defined as follows:

$$\mathcal{L}_{motion} = \sum_{i=1}^{K} \mathbb{1}_{\{m_i \neq \emptyset\}} \mathcal{L}_{seg}(m_i, W^t_{:,\hat{\sigma}(i)}). \tag{2}$$

That is, the loss is only computed for the slots for which motion masks have been assigned, and the remaining slots are not constrained and can bind to any regions in the image. This is illustrated in the right part of Figure 2, where motion segmentation masks are available for only two objects in a crowded outdoor scene, and they get matched to the slots whose attention maps are most similar to the masks. The remaining slots are unconstrained, but still manage to capture both moving and static objects, as well as the background, driven by the image reconstruction objective. The actual segmentation loss $\mathcal{L}_{seg}$ in Eq. 2 is the binary cross entropy:

$$\mathcal{L}_{seg}(m, W) = \sum_{j=1}^{N} -m_j \log(W_j) - (1-m_j) \log(1-W_j). \tag{3}$$

3.4. Loss function and optimization

Our final objective is defined as follows:

$$\mathcal{L} = \mathcal{L}_{recon} + \lambda_M \mathcal{L}_{motion} + \lambda_T \mathcal{L}_{temp}, \quad (4)$$

where $\mathcal{L}_{recon}$ is the MSE loss for the image reconstruction, $\mathcal{L}_{temp}$ is a temporal consistency regularization term, and λ_M and λ_T are the weights for the motion supervision and temporal consistency terms. The latter is defined as

$$\mathcal{L}_{temp}(S) = \sum_{t=1}^{T-1} \|\mathbb{I} - softmax(S^t \cdot (S^{t+1})^{\mathbf{T}})\|, \quad (5)$$

where $\mathbb{I} \in \mathbb{R}^{K \times K}$ is the identity matrix. It is easy to see that this term is encouraging similarity between feature representations of the slots in consecutive frames and thus improving temporal consistency of the slot bindings. The model is trained on video clips of length T and we ensure that at least half of the clips in a batch have a non-empty set of motion segmentations $\mathcal{M}$.

4. Experimental evaluation

4.1. Datasets and evaluation

We use two synthetic datasets for the analysis of the proposed approach: CATER [20] for ablating the architecture of the model and a realistic ParallelDomain (TRI-PD) dataset for analyzing the impact of the motion segmentation quality on the model's performance. In addition, we use a real-world KITTI benchmark [19] for comparison to the state of the art.

CATER is a video version of the CLEVR [30] dataset which was used in many recent works on unsupervised object discovery [8, 29, 38]. We utilize the provided engine to generate 2,000 videos by placing between 4 and 8 geometric shapes, such as cubes or cones, on a plain background at random, and assigning a random color to each instance. Each object can then move on a random trajectory or remain static, and the camera motion is also randomized. We use 1,600 videos for training and 400 for evaluation, with each video being 40-frames long with a resolution 128×128 (see Figure 3, left). For ablation analysis, we randomly assign one object as moving in each video and use the ground truth mask of that object as a motion mask. Notice that we do experiment with automatically estimated motion segmentation on more challenging TRI-PD and KITTI.

ParallelDomain (TRI-PD) is a synthetic dataset with street driving scenarios (see Figure 3, center). It was collected using a state-of-the-art synthetic data generation service [1]. The training set contains 924 photo-realistic, 10 seconds long videos with driving scenarios in city environments captured at 20 FPS. We use 51 videos from a disjoint set of scenes for evaluation. Each video comes with a full set of ground truth annotations, including optical flow, allowing us to conduct a detailed analysis of the impact of the motion segmentation quality on our method's performance. More statistics and qualitative examples are provided in the supplementary.

Figure 3. Frame samples from the video datasets used in our experiments. CATER [20] (left) is a toy, synthetic dataset similar to the ones used in prior works. TRI-PD (center) is a collection of photo-realistic, synthetic videos, which is a major step forward in visual complexity. KITTI [19] (right) is a real-world benchmark with outdoor scenes.

KITTI is a real-world benchmark with city driving scenarios which comes with a variety of annotations (Figure 3, right). In this work, we use the instance segmentation subset of the dataset for evaluation. It contains 200 frames, which we resize to $368 \times 1,248$. Notice that instance segmentation annotations are provided on individual images in this dataset, without the temporal context, allowing us to demonstrate that our approach does not require videos at inference time. Since our model is unsupervised, we use all the 147 videos in the training set of KITTI to discover the objects that can move in the real world.

Evaluation metrics. We use Adjusted Rand Index (ARI) as the main metric for comparing object discovery capabilities of the models, but also report more traditional segmentation metrics like F-measure and mIoU in the supplementary. ARI is a clustering similarity metric which captures how well predicted segmentation masks match ground truth masks in a permutation-invariant fashion. This is more suitable for the evaluation of unsupervised approaches than, say, mIoU, because it does not require for the methods to make the decision which segments represent the objects and which correspond to the background. Following prior work [22, 38], we only measure ARI based on foreground objects, which we refer to as Fg. ARI.

4.2. Implementation details

For the components of our model shared with SlotAttention [38] we follow their architecture and training protocol exactly, and describe the remaining details below.

We replace the shallow encoder used in [38] with a ResNet18 [25] to scale the representational power to realistic scenes. We also experiment with deeper backbones in the supplementary. All the models are trained from scratch unless stated otherwise. We additionally report results with contrastive-learning pre-training in the supplementary. To be able to capture small objects, we remove the last 2 max pooling layers from the ResNet, and add a corresponding

dilation ratio to preserve the field of view. We use 10 slots for the experiments on CATER and 45 slots on TRI-PD and KITTI to account for the larger number of objects.

All the models are trained for 500 epochs using Adam [34] with a batch size 20 and learning rate 0.001. Following [38], we use learning rate warm-up [21] and an exponential decay schedule to prevent early saturation and reduce variance. We set λ_M to 0.5 and λ_T to 0.01 on the validation set of CATER, and use these value in all the experiments. Video-based variants are trained using clips of length 5. At inference time, the model is evaluated in a sliding window fashion with a stride 5.

We experiment with two motion segmentation algorithms – a heuristic-based [33], and a learning-based one [14], for which we only use the motion stream trained on the toy FlyingThings3D dataset [39]. Both methods take optical flow as input, so we evaluate them with both ground truth flow, and flow estimated with the state-of-the-art supervised [55] and unsupervised [53] approaches. Since the outputs of both methods contain many noisy segments, we apply a few generic post-processing steps to clean up the results. They remove very large and very small segments, as well as segments at the image boundary. The details of the post-processing are provided in the supplementary.

We compare our approach to several recent learning-based object discovery algorithms as well as to a classical, heuristic-based method. In particular, we choose SlotAttention [38], MONet [8], SCALOR [29], and S-IODINE [22] as a representative sample of learning-based methods, with S-IODINE also being a video-based approach. For MONet and S-IODINE, we replace the original backbone with ResNet18 and match the input resolution to the one used by our method for a fair comparison, but keep all the other details intact. All the models are trained until convergence. We use MCG [3] as an heuristic-based baseline. It is a proposal generation method, so to obtain a single interpretation of an image, we sample the top scoring proposals until all the pixels are covered. For overlapping segments, we assign the corresponding pixels to the smaller segment.

4.3. Architectural analysis

In this section, we begin the analysis of our method by studying the variants of the auto-encoder framework for object discovery on the validation set of CATER in Table 1. Firstly, we evaluate the original SlotAttention model (row 1 in the table), which serves as a basis for our approach, and find that it performs reasonably well on this toy dataset, though the Fg. ARI scores are noticeable lower than those reported in the original paper [38] on CLEVR. This is explained by the fact that the scenes in CATER are more challenging, with a larger variance in the number of objects and more occlusions.

Next, we convert the frame-level architecture of SlotAttention to a video-level model by adding a ConvGRU after the encoder. This has only a minor effect on the performance when trained on 1-frame sequences (row 2 in the table), but training on video clips (row 3) results in a 5.2 points increase in Fg. ARI score. This demonstrates that the feature space of the recurrent model can capture video dynamics and thus simplify separating objects from the background.

ConvGRU	Slot inf.	Temp.	Decode.	Motion	Recon.	Fg. ARI
–	Iter.	✗	Per slot	✗	✓	64.4
frame	Iter.	✗	Per slot	✗	✓	66.3
clip	Iter.	✗	Per slot	✗	✓	71.5
clip	1-shot	✗	Per slot	✗	✓	83.2
clip	1-shot	✓	Per slot	✗	✓	86.7
clip	1-shot	✓	1-shot	✗	✓	34.5
clip	1-shot	✓	1-shot	✓	✓	**92.7**
clip	1-shot	✓	1-shot	✓	✗	77.9

Table 1. Analysis of the model architecture using Fg. ARI on the validation set of CATER. We ablate the ConvGRU module, slot inference strategy, temporal consistency constraint, decoding strategy, independent motion prior, and the reconstruction objective. Combining motion priors with reconstruction leads to best results.

However, going from single frame inputs to clips increases the memory requirements of the model. To mitigate this issue, we now study the architectural modifications proposed in Section 3.2. Firstly, replacing iterative inference on randomly initialized slots with a single attention operation with a learnable initialization not only results in an improved computational efficiency, but also significantly improves the performance. Incorporating the temporal consistency term in the loss further boosts the Fg. ARI score due to more robust slot binding. Next, switching to 1-shot decoding significantly reduces the memory consumption of the model, but also results in it largely loosing its object-discovery capabilities. This demonstrates that individual slot decoding was crucial for enforcing the spatial cohesion prior to the SlotAttention model.

Despite this disadvantage, incorporating a weak learning signal in the form of a motion segmentation not only recovers, but significantly improves the model's performance. This demonstrates that independent motion is a much stronger and more generic prior than appearance and location similarity used in the SlotAttention, even in a toy dataset like CATER. Finally, the last row of Table 1 shows that the reconstruction objective is still important for achieving top performance by enforcing generalization from moving to static instances.

4.4. Object discovery in realistic videos

We now explore how well the model introduced above scales to realistic outdoor scenes in the TRI-PD dataset in Table 2 and Figure 4. We separately report the Fg. ARI score for moving and static objects to assess the network's generalization abilities. We begin with evaluating the baseline variant of our model without independent motion priors, and observe that appearance similarity is indeed not sufficient for

Model	Motion seg.	Fg. ARI Stat.	Fg. ARI Mov.	Fg. ARI All
Ours	None	10.5	18.4	13.1
Ours	GT all	69.0	72.2	**71.7**
Ours	GT moving	53.3	62.7	59.6
Ours	GT flow + [33]	39.9	47.5	42.8
Ours	GT flow + [14]	48.3	54.9	51.7
Ours	RAFT flow + [14]	46.8	55.6	50.9
Ours	SMURF flow + [14]	47.3	54.8	50.5
-	RAFT flow + [14]	2.7	5.3	3.4

Table 2. Analysis of the effect of the quality of motion segmentation on the model's performance on the validation set of TRI-PD. We gradually reduce the quality of the motion segments starting from ground truth to fully estimated. Our method learns to discover both moving and static instances guided by a very sparse motion signal.

	Learning-based	TRI-PD	KITTI
SlotAttention [38]	✓	10.2	13.8
MONet [8]	✓	11.0	14.9
SCALOR [29]	✓	18.6	21.1
S-IODINE [22]	✓	9.8	14.4
MCG [3]	✗	25.1	40.9
Ours	✓	**50.9**	**47.1**

Table 3. Comparison to the state-of-the-art approaches to object discovery on the validation sets of TRI-PD and KITTI using Fg. ARI. Our approach outperforms both learning- and heuristic-based methods by capitalizing on independent motion cues.

object discovery in realistic scenes, as reflected by the low Fg. ARI score. Qualitatively, the first column of Figure 4 illustrates that this variant completely fails to discover any objects, and instead segments the scene into random patches based on color and location similarity.

Next, we establish the upper bound for our model's performance by using all the ground truth object segments (corresponding to moving and static objects) for training. This fully-supervised approach reaches an Fg. ARI score of 71.7, which is significantly below the 92.7 obtained by the best version of our model on CATER, further emphasizing the complexity of TRI-PD. Qualitatively, as can be seen in the second column of Figure 4, this variant successfully captures all the clearly visible objects in a scene, and also groups the background pixels together.

Only using the ground truth segments corresponding to the moving objects, which simulates the theoretical scenario in which we have a perfect motion segmentation algorithm, does result in a performance drop of 11.3 Fg. ARI points, which is especially noticeable for static objects, but the overall score remains 46.5 points higher than the baselines trained without the motion prior. Qualitatively, the model is able to accurately segment most of the moving and static instances, as shown in the third column in Figure 4. However, this variant oversegments the background, demonstrating that explaining as many objects in the scene as possible is crucial for learning a strong background model.

Switching to actual motion segmentation algorithms, we first compare the state-of-the-art heuristic-based and learning-based methods using the ground truth optical flow as input in rows 5 and 6 of the Table 2. As expected, we observe that the more recent learning-based method produces more accurate motion segmentations, which in turn results in a higher performance of our approach. Qualitatively, this model, shown in the 4th column in Figure 4, has a slightly lower recall than the variant trained with ground truth moving segments due to the sparser learning signal. Intriguingly, replacing ground truth flow with the one estimated with a state-of-the-art supervised RAFT [55], or self-supervised SMURF [53] algorithms barely changes the performance, despite a noticeable decrease in the motion segmentation quality (last column in Figure 4). This result demonstrates the robustness of our method to noise. We use RAFT flow for the remainder of the experiments.

Finally, to better quantify the ability of our model to generalize from sparse, noisy motion segmentations to the whole distribution of objects in crowded scenes, we evaluate the Fg. ARI score of the motion segmentations themselves in the last row of Table 2. We can see that these masks indeed mostly capture the moving objects; however, even for those, only a tiny fraction is segmented. In contrast, our approach, capitalizing on this noisy and incomplete signal, increases the overall ARI score by a factor of 15.

4.5. Comparison to the state of the art

Finally, we compare our approach to the state of the art on the validation sets of TRI-PD and KITTI in Table 3. Firstly, we observe that all the learning-based methods fail to achieve non-trivial results on both datasets. This confirms our hypothesis that appearance alone is not a sufficient signal to separate objects from the background in realistic environments. In contrast, our proposed approach outperforms all these methods by a wide margin by capitalizing on independent motion cues.

Interestingly, the classical MCG approach performs significantly better than the more recent learning-based methods (moreover, this observation holds even on the toy CATER benchmark, as we show in the supplementary). Our method outperforms MCG on both datasets, with the margin being significantly larger on TRI-PD. Recall that KITTI is an image-based benchmark, where the annotated frames are selected to prominently feature the objects of interest. In contrast, TRI-PD is a densely labeled video dataset with more challenging camera angles and more background clutter (see Figure 5 for a qualitative comparison). Thus, wider margins on PD highlight the benefits of our learning-based approach compared to the heuristic-based MCG.

5. Discussion and limitations

Discovering objects and their extent from raw data is a challenging problem due to the ambiguity of what constitutes

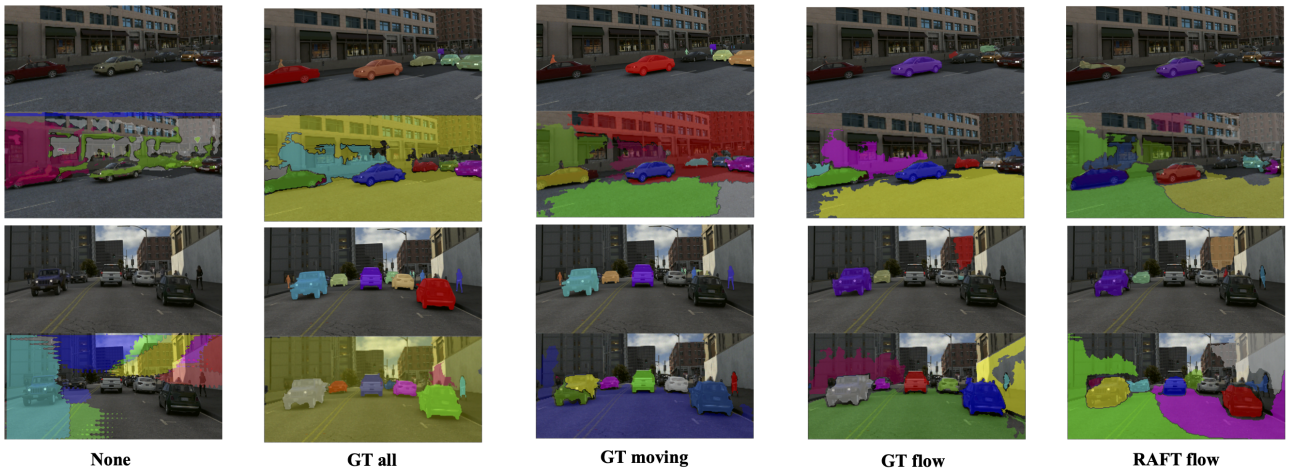

Figure 4. Top-10 masks produced by our model with varying quality of motion priors on the validation set of TRI-PD. We show the motion masks used for supervision on top of the corresponding model's outputs. In the last two columns the approach of Dave et al. [14] is used for motion segmentation. Our method learns to discover the objects even with sparse and noisy motion segmentation based on estimated flow.

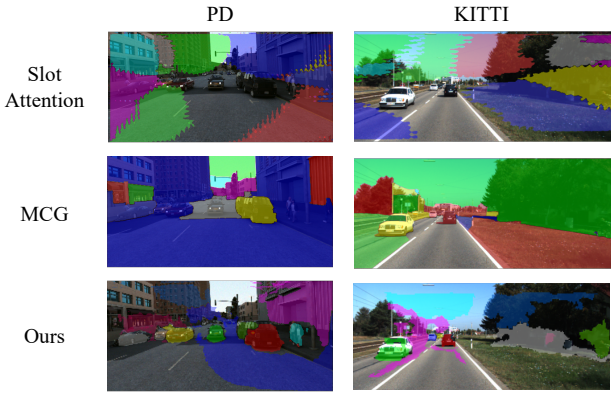

Figure 5. Qualitative comparison of our approach and representative heuristic- and learning-based methods on the validation sets of TRI-PD and KITTI (showing top-10 masks). Ours learns to successfully separate objects from the background, whereas appearance-based methods struggle in cluttered environments.

an object. In this work, we propose one way to automatically resolve this ambiguity by focusing on dynamic objects and using independent motion as an inductive bias in an auto-encoder framework. Our analysis demonstrates promising results in real-world environments, while further raising a number of important questions.

Generalization to non-dynamic objects. While independent object motion provides a convenient signal for object discovery from data, it ignores objects that are not capable of moving by themselves, but might be important for downstream tasks. In particular, in indoor environments people interact with accessories, electronics, food, etc., and capturing these objects is crucial for action recognition [45, 66] and robotics [6, 40]. Notice, however, that extending the definition of a dynamic object to those entities that either move by themselves or can be moved by humans covers most of such cases. Classical motion segmentation approaches [7, 33] do attempt to capture all the objects that fall into this more general definition, but do not generalize in the wild. Developing more robust, learning-based versions of these methods is a critical step towards a generic object discovery algorithm.

Object category imbalance in the real world. Like any other learning-based method, ours is susceptible to focusing on the most common categories, while ignoring the objects in the tail of the distribution. For instance, in the real world we might see lots of moving people, vehicles and animals, and sometimes a person picking up a piece of litter. In theory, this should allow our method to discover not only what people, cars and animals are, but also litter. However, it might happen too infrequently in practice. Fortunately, this problem has received a lot of attention in the few-shot and continual learning domains [10, 32, 48, 65], and the proposed solutions can be integrated into our framework.

Supervision used to train the motion segmentation algorithm. The approach of Dave et al. [14], used in our experiments, is trained on the toy, synthetic FlyingThings3D [39] dataset with ground truth moving object masks. This raises the question of whether it is this indirect object-level supervision which makes our method outperform other, fully unsupervised approaches. To address this concern, in the supplementary we directly pre-train SlotAttention on FlyingThings3D in a fully-supervised way, showing this does not have a significant effect on its object discovery performance in realistic videos due to the large domain gap.

Acknowledgements. We thank Alexei Efros, Vitor Guizilini and Jie Li for their valuable comments, and Achal Dave for his help with computing motion segmentations. This research was supported by Toyota Research Institute.

References

[1] Parallel domain. https://paralleldomain.com/, November 2021. 2, 5

[2] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *CVPR*, 2015. 2, 3

[3] Pablo Arbeláez, Jordi Pont-Tuset, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping. In *CVPR*, 2014. 1, 2, 6, 7

[4] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. In *ICLR*, 2016. 2, 3

[5] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemyslaw Debiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019. 1

[6] Aude Billard and Danica Kragic. Trends and challenges in robot manipulation. *Science*, 364(6446), 2019. 8

[7] Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. In *ECCV*, 2010. 2, 8

[8] Christopher P Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. MONet: Unsupervised scene decomposition and representation. *arXiv preprint arXiv:1901.11390*, 2019. 1, 2, 3, 5, 6, 7

[9] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1, 3, 4

[10] Nadine Chang, Zhiding Yu, Yu-Xiong Wang, Anima Anandkumar, Sanja Fidler, and Jose M Alvarez. Image-level or object-level? A tale of two resampling strategies for long-tailed detection. In *ICML*, 2021. 8

[11] Chang Chen, Fei Deng, and Sungjin Ahn. Roots: Object-centric representation and rendering of 3D scenes. *JMLR*, 2021. 2

[12] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*, 2014. 3

[13] M. Cynader, N. Berman, and A. Hein. Cats reared in stroboscopic illumination: Effects on receptive fields in visual cortex. *Proceedings of the National Academy of Sciences*, 70(5):1353–1354, 1973. 2

[14] Achal Dave, Pavel Tokmakov, and Deva Ramanan. Towards segmenting anything that moves. In *ICCV Workshops*, 2019. 1, 2, 6, 7, 8

[15] Yilun Du, Kevin Smith, Tomer Ulman, Joshua Tenenbaum, and Jiajun Wu. Unsupervised discovery of 3D physical objects from video. In *ICLR*, 2021. 2

[16] Martin Engelcke, Adam R Kosiorek, Oiwi Parker Jones, and Ingmar Posner. GENESIS: Generative scene inference and sampling with object-centric latent representations. In *ICLR*, 2020. 1, 2

[17] SM Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, Geoffrey E Hinton, et al. Attend, infer, repeat: Fast scene understanding with generative models. In *NeurIPS*, 2016. 2

[18] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004. 1, 2

[19] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *CVPR*, 2012. 2, 5

[20] Rohit Girdhar and Deva Ramanan. CATER: A diagnostic dataset for compositional actions and temporal reasoning. In *ICLR*, 2020. 5

[21] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: Training ImageNet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 6

[22] Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. In *ICML*, 2019. 1, 2, 3, 5, 6, 7

[23] Klaus Greff, Antti Rasmus, Mathias Berglund, Tele Hao, Harri Valpola, and Jürgen Schmidhuber. Tagger: Deep unsupervised perceptual grouping. In *NeurIPS*, 2016. 2

[24] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017. 1, 2

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, 2016. 5

[26] Paul Henderson and Christoph H Lampert. Unsupervised object-centric video generation and decomposition in 3D. In *NeurIPS*, 2020. 2

[27] Seunghoon Hong, Donghun Yeo, Suha Kwak, Honglak Lee, and Bohyung Han. Weakly supervised semantic segmentation using web-crawled videos. In *CVPR*, 2017. 2, 3

[28] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*, 2017. 2

[29] Jindong Jiang, Sepehr Janghorbani, Gerard De Melo, and Sungjin Ahn. SCALOR: Generative world models with scalable object representations. In *ICLR*, 2020. 2, 5, 6, 7

[30] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017. 5

[31] Daniel Kahneman, Anne Treisman, and Brian J Gibbs. The reviewing of object files: Object-specific integration of information. *Cognitive psychology*, 1992. 1

[32] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *ICLR*, 2020. 8

[33] Margret Keuper, Bjoern Andres, and Thomas Brox. Motion trajectory segmentation via minimum cost multicuts. In *ICCV*, 2015. 2, 6, 7, 8

[34] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[35] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014. 2

[36] Kurt Koffka. *Principles of Gestalt psychology*. Routledge, 2013. 2

[37] Zhixuan Lin, Yi-Fu Wu, Skand Vishwanath Peri, Weihao Sun, Gautam Singh, Fei Deng, Jindong Jiang, and Sungjin Ahn. Space: Unsupervised object-oriented scene representation via spatial attention and decomposition. In *ICLR*, 2020. 1, 2

[38] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *NeurIPS*, 2020. 1, 2, 3, 4, 5, 6, 7

[39] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016. 3, 6, 8

[40] Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 6-dof graspnet: Variational grasp generation for object manipulation. In *ICCV*, 2019. 8

[41] Peter Ochs and Thomas Brox. Higher order motion models and spectral clustering. In *CVPR*, 2012. 2

[42] Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *ICCV*, 2013. 3

[43] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *CVPR*, 2017. 2, 3

[44] Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012. 2, 3

[45] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *ECCV*, 2018. 8

[46] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 1

[47] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 2014. 2

[48] Dvir Samuel and Gal Chechik. Distributional robustness loss for long-tail learning. In *ICCV*, 2021. 8

[49] Elizabeth S Spelke. Principles of object perception. *Cognitive science*, 1990. 2, 3

[50] Elizabeth S Spelke and Katherine D Kinzler. Core knowledge. *Developmental science*, 2007. 1

[51] Karl Stelzner, Kristian Kersting, and Adam R Kosiorek. Decomposing 3D scenes into objects via unsupervised volume segmentation. *arXiv preprint arXiv:2104.01148*, 2021. 2

[52] Russell Stewart, Mykhaylo Andriluka, and Andrew Y Ng. End-to-end people detection in crowded scenes. In *CVPR*, 2016. 4

[53] Austin Stone, Daniel Maurer, Alper Ayvaci, Anelia Angelova, and Rico Jonschkowski. SMURF: Self-teaching multi-frame unsupervised RAFT with full-image warping. In *CVPR*, 2021. 2, 6, 7

[54] Matthias Tangemann, Steffen Schneider, Julius von Kügelgen, Francesco Locatello, Peter Gehler, Thomas Brox, Matthias Kümmerer, Matthias Bethge, and Bernhard Schölkopf. Unsupervised object learning via common fate. *arXiv preprint arXiv:2110.06562*, 2021. 3

[55] Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020. 2, 6, 7

[56] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Weakly-supervised semantic segmentation using motion cues. In *ECCV*, 2016. 2, 3

[57] Pavel Tokmakov, Cordelia Schmid, and Karteek Alahari. Learning to segment moving objects. *IJCV*, 2019. 3

[58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3

[59] Rishi Veerapaneni, John D Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua Tenenbaum, and Sergey Levine. Entity abstraction in visual model-based reinforcement learning. In *CoRL*, 2020. 2

[60] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019. 1

[61] Christopher Xie, Yu Xiang, Zaid Harchaoui, and Dieter Fox. Object discovery in videos as foreground motion clustering. In *CVPR*, 2019. 2

[62] Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. Self-supervised video object segmentation by motion grouping. *arXiv preprint arXiv:2104.07658*, 2021. 3

[63] Yanchao Yang, Brian Lai, and Stefano Soatto. DyStaB: Unsupervised object segmentation via dynamic-static bootstrapping. In *CVPR*, 2021. 3

[64] Peiyu Yu, Sirui Xie, Xiaojian Ma, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. Unsupervised foreground extraction via deep region competition. In *NeurIPS*, 2021. 2

[65] Xing Zhang, Zuxuan Wu, Zejia Weng, Huazhu Fu, Jingjing Chen, Yu-Gang Jiang, and Larry Davis. VideoLT: Large-scale long-tailed video recognition. In *ICCV*, 2021. 8

[66] Yubo Zhang, Pavel Tokmakov, Martial Hebert, and Cordelia Schmid. A structured model for action detection. In *CVPR*, 2019. 8