

Unsupervised Segmentation in Real-World Images via Spelke Object Inference

Honglin Chen¹, Rahul Venkatesh¹, Yoni Friedman⁴, Jiajun Wu¹,
Joshua B. Tenenbaum⁴, Daniel L. K. Yamins^{1,2,3,**}, and Daniel M. Bear^{2,3,**}

¹ Department of Computer Science, Stanford

² Department of Psychology, Stanford

³ Wu Tsai Neurosciences Institute, Stanford

⁴ Department of Brain and Cognitive Sciences and CBMM, MIT

{honglinc,dbear}@stanford.edu

** = Equal contribution

Abstract. Self-supervised, category-agnostic segmentation of real-world images is a challenging open problem in computer vision. Here, we show how to learn static grouping priors from motion self-supervision by building on the cognitive science concept of a Spelke Object: a set of physical stuff that moves together. We introduce the Excitatory-Inhibitory Segment Extraction Network (EISEN), which learns to extract pairwise affinity graphs for static scenes from motion-based training signals. EISEN then produces segments from affinities using a novel graph propagation and competition network. During training, objects that undergo correlated motion (such as robot arms and the objects they move) are decoupled by a bootstrapping process: EISEN explains away the motion of objects it has already learned to segment. We show that EISEN achieves a substantial improvement in the state of the art for self-supervised image segmentation on challenging synthetic and real-world robotics datasets.

1 Introduction

Most approaches to image segmentation rely heavily on supervised data that is challenging to obtain and are largely trained in a category-specific way [31, 14, 17, 6]. Thus, even state of the art segmentation networks struggle with recognizing untrained object categories and complex configurations [10]. A self-supervised, category-agnostic segmentation algorithm would be of great value.

But how can a learning signal for such an algorithm be obtained? The cognitive science of perception in babies provides a clue, via the concept of a *Spelke object* [33]: a collection of physical stuff that moves together under the application of everyday physical actions. Perception of Spelke objects is category-agnostic and acquired by infants without supervision [33]. In this work we build

² More formally, two pieces of stuff are considered to be in the same Spelke object if and only if, under the application of any sequence of actions that causes sustained motion of one of the pieces of stuff, the magnitude of the motion that the other piece of stuff experiences relative to the first piece is approximately zero compared to the

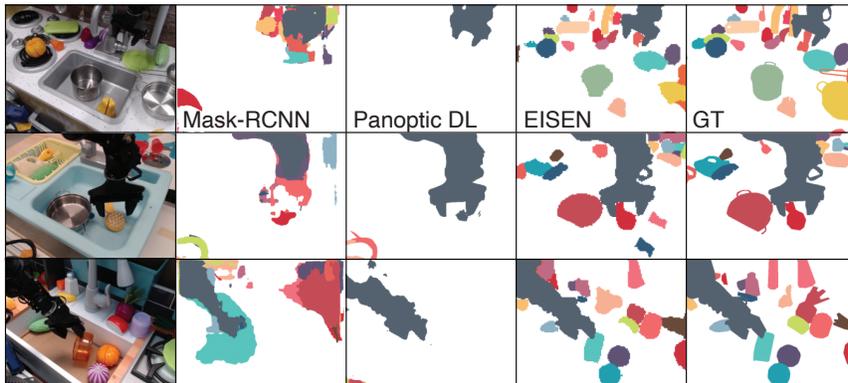


Fig. 1: **Unsupervised Segmentation of Spelke Objects.** Two standard object segmentation architectures, Mask-RCNN and Panoptic DeepLab, largely fail to learn to detect Spelke objects in the **Bridge** dataset without dense, categorical supervision. In contrast, our approach (EISEN) can detect these objects, without any supervision, via motion-based bootstrapping: learning to predict what moves together, then using top-down inference to segregate arm from object motion.

a neural network that learns from motion signals to segment Spelke objects in still images (Fig. 1). To achieve this goal, we make two basic innovations.

First, we design a pairwise affinity-based grouping architecture that is optimized for learning from motion signals. Most modern segmentation networks are based on pixelwise background-foreground categorization [17, 6]. However, Spelke objects are fundamentally relational, in that they represent whether *pairs* of scene elements are likely to move together. Moreover, this physical connectivity must be learned from real-world video data in which motion is comparatively sparse, as only one or a few Spelke objects is typically moving at a time (Fig. 2, top). Standard pixelwise classification problems that attempt to approximate these pairwise statistics (such as the “Spelke-object-or-not” task) induce large numbers of false negatives for temporarily non-moving objects. Directly learning pairwise affinities avoids these problems.

To convert affinities into actual segmentations, we implement a fully differentiable grouping network inspired by the neuroscience concepts of recurrent label propagation and border ownership cells [28, 41]. This network consists of (i) a quasi-local, recurrent affinity Propagation step that creates (soft) segment identities across pixels in an image and (ii) a winner-take-all Competition step that assigns a unique group label to each segment. We find through ablation studies that this specific grouping mechanism yields high-quality segments from affinities.

A second innovation is an iterative scheme for network training. In real-world video, most objects are inanimate, and thus only seen in motion when caused to

magnitude of overall motion. Natural action groups arise from the set of all force applications exorable by specific physical actuator, such as (e.g.) a pair of human hands or a robotic gripper.

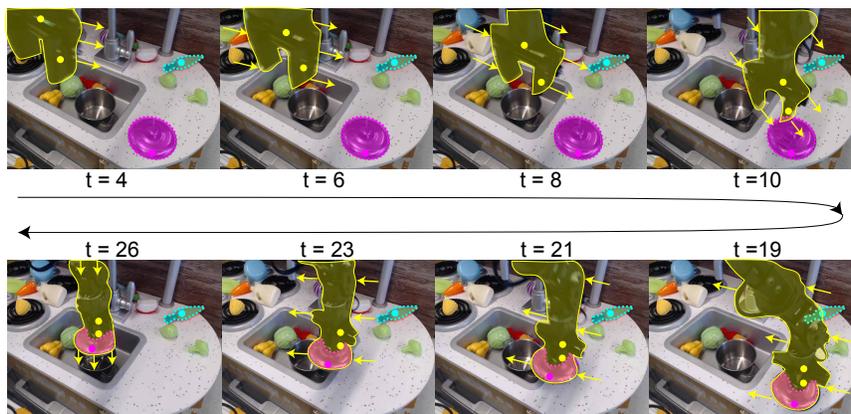


Fig. 2: **Two Challenges of Learning Spelke Objects.** (**Top Row**) Motion in real-world video is sparse. Thus, pairwise inferences about whether two points are moving together (e.g. the yellow points on the robot arm) or are not moving together (e.g. any yellow-cyan pairing) are valid. However, pointwise motion-based inferences of whether a point is in a Spelke object or not will have many false negatives (e.g. points in the cyan object). (**Bottom row**) Inanimate objects (e.g. magenta lid) only move when moved by something else (e.g. the robotic arm), requiring explaining away of the apparent motion correlation (e.g. yellow-magenta pairs in the bottom row).

move by some other animate object, such as a human hand or robotic gripper (Fig 2, bottom). This correlated motion must therefore be dissociated to learn to segment the mover from the moved object. Cognitive science again gives a clue to how this may be done: babies first learn to localize hands and arms, then later come to understand external objects [38]. We implement this concept as a confidence-thresholded bootstrapping procedure: motion signals that are already well-segmented by one iteration of network training are explained away, leaving unexplained motions to be treated as independent sources for supervising the next network iteration. For example, in natural video datasets with robotic grippers, the gripper arm will naturally arise as a high-confidence segment first, allowing for the object in the gripper to be recognized as a separate object via explaining-away. The outputs of this explaining away train the next network iteration to recognize inanimate-but-occasionally-moved objects in still images, even when they not themselves being moved.

We train this architecture on optical flow from unlabeled real-world video datasets, producing a network that estimates high-quality Spelke-object segmentations on still images drawn from such videos. We call the resulting network the Excitatory-Inhibitory Segment Extraction Network (EISEN). We show EISEN to be robust even when the objects and configurations in the training videos and test images are distinct. In what follows, we review the literature on related works, describe the EISEN architecture and training methods in detail, show results on both complex synthetic datasets and real-world videos, and analyze algorithmic properties and ablations.

2 Related Work

Segmentation as bottom-up perceptual grouping. The Gestalt psychologists discovered principles according to which humans group together elements of a scene, such as feature similarity, boundary closure, and correlated motion (“common fate”) [36]. This inspired classical computer vision efforts to solve segmentation as a bottom-up graph clustering problem [31, 26]. Although these approaches achieved partial success, they have proved difficult to adapt to the enormous variety of objects encountered in real-world scenes like robotics environments. Thus, today’s most successful algorithms instead aim to segment objects by learning category-specific cues on large, labeled datasets [17, 6, 42].

Unsupervised and category-agnostic segmentation. Several recent approaches have tried to dispense with supervision by drawing on advances in self-supervised object *categorization*. DINO, LOST, and Token-Cut perform “object discovery” by manipulating the attention maps of self-supervised Vision Transformers, which can be considered as maps of affinity between an image patch and the rest of the scene [5, 32, 39]. PiCIE learns to group pixels in an unsupervised way by encouraging particular invariances and equivariances across image transformations. While these early results are encouraging, they apply more naturally to *semantic* segmentation than to grouping individual Spelke objects (instance segmentation): to date, they are mostly limited either to detecting a single object per image or to grouping together all the objects of each category. The GLOM proposal [19] sketches out an unsupervised approach for constructing “islands” of features to represent object parts or wholes, which is similar to our grouping mechanism; but it does not provide a specific algorithmic implementation. We find the architectural particulars of EISEN are essential for successful object segmentation in real-world images (see **Ablations**).

Object discovery from motion. A number of unsupervised object discovery methods can segment relatively simple synthetic objects but struggle on realistic scenes [15, 24, 8, 20]. When applied to the task of *video object segmentation*, Slot Attention-based architectures can segment realistic moving objects [21, 40], but none of these methods uses motion to learn to segment the majority of objects that are static at any given time. Several approaches discover objects via motion signals, making a similar argument to ours for motion revealing physical structure [30, 8, 1, 34, 37, 7, 29]. However, they have been limited to segmenting a narrow range of objects or scenes.

We hypothesize that generalization to realistic, complex scenes benefits greatly from affinity-based grouping and learning. In this respect, our work is heavily inspired by PSGNet, an unsupervised affinity-based network that learns to segment scenes from both object motion and other grouping principles [2]. We make two critical advances on that work: (1) replacing its problematic (and non-differentiable) Label Propagation algorithm with a neural network; and (2) introducing a bootstrapping procedure that uses top-down inference to explain raw motion observations in terms of confidently grouped objects. In combination, these novel contributions allow EISEN to accurately perform a challenging task: the static segmentation of real-world objects without supervision.

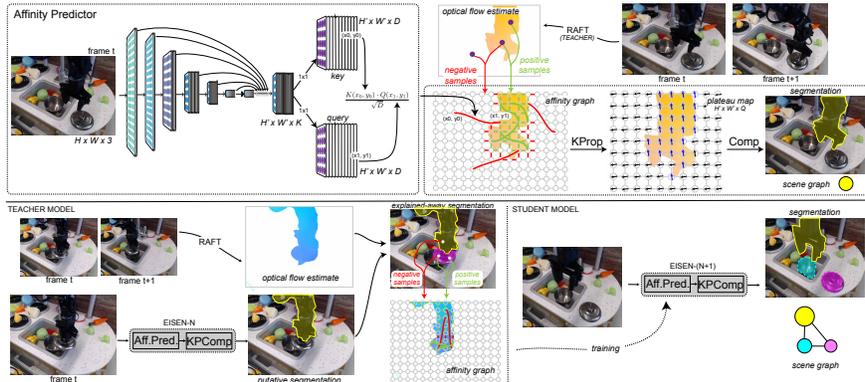


Fig. 3: The EISEN architecture and training process. (Top: Architecture) The EISEN architecture consists of (i) an Affinity Predictor module which extracts a pairwise affinity graph for each scene, and (ii) the KProp-Competition module, which converts the affinity graph into an actual segmentation. The affinity predictor is trained to predict thresholded optical flow estimates computed via the RAFT algorithm, with positive samples corresponding to pairs of moving points (green affinity graph edges), and negative samples corresponding to moving-nonmoving point pairs (red edges). Edges are computed for all pairs of close-by points and a sampling of further-separated point pairs. Segments are extracted from the affinity graph via a two-stage mechanism consisting of Kaleidoscopic Propagation and inter-node Competition (see text and Fig. 4 for more details). **(Bottom: Iterative Training)** Differences between RAFT optical flow estimates and high-confidence segments from static stage- N EISEN outputs are “explained away” by positing the existence of new Spelke objects, which are then used to supervise the stage- $(N + 1)$ EISEN model.

3 Methods

3.1 The EISEN Architecture

EISEN performs *unsupervised, category-agnostic segmentation of static scenes*: it takes in a single $H \times W \times 3$ RGB image and outputs a segmentation map of shape $H' \times W'$. EISEN and baseline models are trained on the optical flow predictions of a RAFT network [35] pretrained on Sintel [3]. RAFT takes in a pair of frames, so EISEN requires videos for training but not inference.

Overall concept. The basic idea behind EISEN is to construct a high-dimensional feature representation of a scene (of shape $H' \times W' \times Q$) that is almost trivial to segment. In this desired representation, all the feature vectors \mathbf{q}_{ij} that belong to the same object are aligned (i.e., have cosine similarity ≈ 1) and all feature vectors that belong to distinct objects are nearly orthogonal (cosine similarity ≈ 0). A spatial slice of this feature map looks like a set of flat object segment “plateaus,” so we call it the *plateau map* representation. Object segments can be extracted from a plateau map by finding clusters of vectors pointing in similar directions.

The plateau map is inherently relational: both building and extracting segments from it are straightforward given accurate pairwise affinities between scene elements. EISEN therefore consists of three modules applied sequentially to a convolutional feature extractor backbone (Figure 3):

1. *Affinity Prediction*, which computes pairwise affinities between features;
2. *Kaleidoscopic Propagation* (KProp), a graph RNN that aligns the vectors of a plateau map by passing messages on the extracted affinity graph;
3. *Competition*, an RNN that imposes winner-take-all dynamics on the plateau map to extract object segments and suppress redundant activity.

All three modules are differentiable, but only Affinity Prediction has trainable parameters. We use the ResNet50-DeepLab backbone in Panoptic-DeepLab[6], which produces output features of shape $H/4 \times W/4 \times 128$.

Affinity Prediction. This module computes affinities $A(i, j, i', j')$ between pairs of extracted feature vectors $\mathbf{f}_{ij}, \mathbf{f}_{i'j'}$. Each feature vector is embedded in \mathbb{R}^D with linear key and query functions, and the affinities are given by standard softmax self-attention plus row-wise normalization:

$$\tilde{A}_{i'j'}^{ij} = \text{Softmax} \left(\frac{1}{\sqrt{D}} (W_k \mathbf{f}_{ij})(W_q \mathbf{f}_{i'j'})^T \right), \quad A_{i'j'}^{ij} = \tilde{A}_{i'j'}^{ij} / \max_{\{i', j'\}} \tilde{A}_{i'j'}^{ij}. \quad (1)$$

To save memory, we typically compute affinities only within a 25×25 grid around each feature vector plus a random sample of long-range “global” affinities.

Kaleidoscopic Propagation. The KProp graph RNN (Figure 4) is a smooth relaxation of the discrete Label Propagation (LProp) algorithm [16]. Besides being nondifferentiable, LProp suffers from a “label clashing” problem: once a cluster forms, the discreteness of labels makes it hard for another cluster to merge with it. This is pernicious when applied to image graphs, as the equilibrium clusters are more like superpixels than object masks [2]. KProp is adapted to the specific demands of image segmentation through the following changes:

- Instead of integers, each node is labeled with a continuous vector $\mathbf{q}_{ij} \in \mathbb{R}^Q$; the full hidden state at iteration s is $h_s \in \mathbb{R}^{N \times Q}$.
- The nondifferentiable message passing in LProp is replaced with two smooth operations: each node sends (1) *excitatory* messages to its high-affinity neighbors, which encourages groups of connected nodes to align; and (2) *inhibitory* messages to its low-affinity neighbors, which orthogonalizes disconnected node pairs. These messages cause clusters of nodes to merge, split, and shift in a pattern reminiscent of a kaleidoscope, giving the algorithm its name.
- At each iteration, node vectors are rectified and ℓ^2 normalized. Although softmax normalization produces (soft) one-hot labels, it reinstates “label clashing” by making the Q plateau map channels compete. ℓ^2 normalization instead allows connected nodes to converge on an intermediate value.

During propagation, the affinity matrix is broken into two matrices, A^+, A^- , for excitatory and inhibitory message passing, respectively. These are simply the original affinity matrix with all values above (resp., below) 0.5 set to zero, then

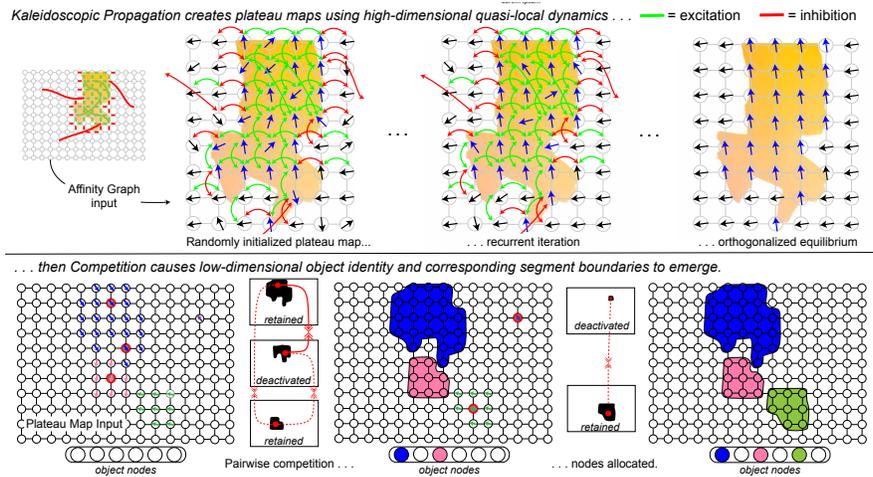


Fig. 4: **Kaleidoscopic Propagation and Competition.** (Top Row: **KProp**) For each node in an input affinity graph, a random normalized Q -dimensional vector is allocated (blue and black arrows). The KProp module is a graph RNN that implements high-dimensional quasi-local dynamics, with affinities A corresponding to excitatory connections and inverted affinities $1 - A$ corresponding to inhibitory connections. These dynamics are repeated a fixed number iterations, quickly coming to equilibrium at a “plateau map” in which candidate segments correspond to nearly-orthogonal domains in the Q -dimensional vector field each of which is nearly-flat. (Bottom Row: **Competition**) Plateau maps are converted into segmentations by having “object nodes” compete for ownership of points within the plateau map. A set of putative object nodes are initialized with randomly located basepoints (red highlighted nodes). Each object node corresponds to an object mask consisting of plateau map locations with high Q -vector correlation to the vector at the basepoint. Pairs of object nodes with overlapping masks compete, with the overall-more-aligned node winning and suppressing alternates. Reinitialization then occurs only over non-covered territory. After a small number of iterations, the process equilibrates with the masks containing segment estimates, and the object nodes describing the scene graph.

normalized by the sum of each row. The plateau map h_0 is randomly initialized and for each of S iterations is updated by

$$h_s^+ = h_s + A^+ h_s, \quad (2)$$

$$h_s^- = h_s^+ - A^- h_s^+, \quad (3)$$

$$h_{s+1} = \text{Norm}(\text{ReLU}(h_s^-)), \quad (4)$$

where Norm does ℓ^2 normalization. We find that convergence is faster if only one random node passes messages at the first iteration.

Competition. Vector clusters in the final plateau map are generic points on the $(Q - 1)$ -sphere, not the (soft) one-hot labels desired of a segmentation map. The Competition module resolves this by identifying well-formed clusters, converting them to discrete *object nodes*, and suppressing redundant activity (Figure 4 bottom.) First, K *object pointers* $\{p^k\} \in \mathbb{R}^2$ are randomly placed

at (h, w) locations on the plateau map and assigned *object vectors* $\mathbf{p}^k \in \mathbb{R}^Q$ according to their positions; an *object segment* $m^k \in \mathbb{R}^{H \times W}$ for each vector is then given by its cosine similarity with the full plateau map:

$$p^k = (p_h^k, p_w^k), \quad \mathbf{p}^k = h_S(p_h^k, p_w^k), \quad m^k = h_S \cdot \mathbf{p}^k. \quad (5)$$

Some of the masks may overlap, and some regions of the plateau map may not be covered by any mask. We use recurrent winner-take-all dynamics to select a minimal set of object nodes that fully explains the map. Let $\mathcal{J}(\cdot, \cdot)$ denote the Jaccard index and let θ be a threshold hyperparameter (set at 0.2 in all our experiments). Competition occurs between each pair of object vectors with masks satisfying $\mathcal{J}(m^k, m^{k'}) > \theta$; the winner is the vector with greater total mask weight $\sum_{i,j} m^k$. An object that wins every pairwise competition is *retained*, while all others are *deactivated* by setting their masks to zero (Figure 4 bottom.) This process is repeated for a total of R iterations by re-initializing each deactivated object $(p^l, \mathbf{p}^l, m^l = 0)$ on parts of the plateau map that remain uncovered, $U = 1 - \sum_k m^k$. Thus the Competition module retains a set of $M \leq K$ nonzero (soft) masks, which are then softmax-normalized along the M dimension to convert them into a one-hot pixelwise segmentation of the scene.

3.2 Training EISEN via Spelke Object Inference

Because KProp and Competition have no trainable parameters, training EISEN is tantamount to training the affinity matrix A . This is done with a single loss function: the row-wise KL divergence between A and a *target connectivity matrix*, \mathcal{C} , restricted to the node pairs determined by *loss mask*, \mathcal{D} :

$$\mathcal{L}_{\text{EISEN}} = \sum_{i,j} \text{KLDiv}(\mathcal{D}_{i'j'}^{ij} \odot A_{i'j'}^{ij}, \mathcal{D}_{i'j'}^{ij} \odot \mathcal{C}_{i'j'}^{ij}). \quad (6)$$

To compute \mathcal{C} and \mathcal{D} we consider pairs of scene elements (a, b) that project into image coordinates (i, j) and (i', j') , respectively. If only one element of the pair is moving (over long enough time scales), it is likely the two elements do not belong to the same Spelke object; when neither is moving, there is no information about their connectivity, so no loss should be computed on this pair. This is the core physical logic – “Spelke object inference” – by which we train EISEN.

Computing connectivity targets from motion. Let $\mathcal{I}(\cdot)$ be a motion indicator function, here $\mathcal{I}(a) = (|\mathbf{flow}_{ij}| > 0)$, where \mathbf{flow} is a map of optical flow. The logic above dictates setting

$$\mathcal{C}_{i'j'}^{ij} \leftarrow 0 \text{ if } (\mathcal{I}(a) \text{ xor } \mathcal{I}(b)), \quad (7)$$

$$\mathcal{D}_{i'j'}^{ij} \leftarrow 1 \text{ if } (\mathcal{I}(a) \text{ or } \mathcal{I}(b)) \text{ else } 0. \quad (8)$$

To learn accurate affinities there must also pairs with $\mathcal{C}_{i'j'}^{ij} = 1$ that indicate when two scene elements belong to the same object.⁵ When a scene contains

⁵ If scenes are assumed to have at most one independent motion source, these are simply the pairs with $\mathcal{I}(a) == \mathcal{I}(b) == 1$. This often holds in robotics scenes (and

multiple uncorrelated motion sources, the optical flow map has an appearance similar to a plateau map (e.g. Figure S2, second column.) This allows the flow map to be segmented into multiple motion sources as if it *were* a plateau map using the Competition algorithm (see Supplement for details.) The positive pairs in the connectivity target can then be set according to

$$\tilde{C}_{i'j'}^{ij} \leftarrow 1 \text{ if } (\mathcal{S}_M(a) == \mathcal{S}_M(b)) \text{ and } (\mathcal{I}(a) == \mathcal{I}(b) == 1) \text{ else } 0, \quad (9)$$

where \mathcal{S}_M is the estimated map of motion segments. Any elements of the background are assumed to be static with $\mathcal{S}_M(a) == \mathcal{I}(a) == 0$ (see Supplement.)

Segmenting correlated motion sources by top-down inference. Naïve application of Equation (9) cannot handle the case of an agent moving a Spelke object (as in Figure 2) because agent and object will be moving in concert and thus will appear as a single “flow plateau.” However, a *non-naïve observer* might have already seen the agent alone moving and have learned to segment it via static cues. If this were so, the agent’s pixels could be “explained away” from the raw motion signal, isolating the Spelke object as its own target segment (Figure 3, lower panels.) Concretely, let $\mathcal{S}_{\mathcal{T}}$ be a map of *confidently segmented objects* output by a teacher model, \mathcal{T} (see Supplement for how EISEN computes confident segments.) Any scene elements that do not project to confident segments have $\mathcal{S}_{\mathcal{T}}(a) = 0$. Then the final loss mask is modified to include all pairs with at least one moving **or** confidently segmented scene element,

$$\hat{D}_{i'j'}^{ij} \leftarrow 1 \text{ if } ((\mathcal{S}_M(a) + \mathcal{S}_{\mathcal{T}}(a) > 0) \text{ or } (\mathcal{S}_M(b) + \mathcal{S}_{\mathcal{T}}(b) > 0)) \text{ else } 0. \quad (10)$$

Explaining away is performed by overwriting pairs in Equation (9) according to whether two scene elements belong to the same or different confident segments, *regardless of whether they belong to the same motion segment*:

$$\hat{C}_{i'j'}^{ij} \leftarrow (\mathcal{S}_{\mathcal{T}}(a) == \mathcal{S}_{\mathcal{T}}(b)) \text{ if } (\mathcal{S}_{\mathcal{T}}(a) + \mathcal{S}_{\mathcal{T}}(b) > 0) \text{ else } \tilde{C}_{i'j'}^{ij}. \quad (11)$$

Thus the final connectivity target, \hat{C} , combines Spelke object inference with the confident teacher predictions, defaulting to the latter in case of conflict.

Bootstrapping. Since objects that appear moving more often should be confidently segmented earlier in training, it is natural to *bootstrap*, using one (frozen) EISEN model as teacher for another student EISEN (Figure 3.) After some amount of training, the student is frozen and becomes the teacher for the next round, as a new student is initialized with the final weights of the prior round. Although bootstrapping could be continued indefinitely, we find that EISEN confidently segments the majority of Spelke objects after three rounds.

4 Results

4.1 Datasets, Training, and Evaluation

Full details of datasets, training, and evaluation are in the Supplement. Briefly, we train EISEN and baseline models on motion signals from three datasets:

is perhaps the norm in a baby’s early visual experience) but not in many standard datasets (e.g. busy street scenes.) We therefore handle the more general case.

Table 1: **Performance (mIoU) of instance segmentation models on the TDW-Playroom dataset.** Models with **full** supervision receive masks for all movable objects in the scene at training time; models with **motion** supervision receive the optical flow predicted by RAFT.

Model	Full supervision		Motion supervision	
	val	test	val	test
SSAP	0.802	0.575	0.295	0.235
DETR	0.860	0.647	0.297	0.258
Panoptic DeepLab	0.870	0.608	0.620	0.373
Mask-RCNN	0.713	0.387	0.629	0.467
EISEN	0.788	0.675	0.730	0.638

Playroom, a ThreeDWorld [12] dataset of realistically simulated and rendered objects (2000 total) that are invisibly pushed; the **DAVIS2016** [27] video object segmentation dataset, which we repurpose to test *static* segmentation learning in the presence of background motion; and **Bridge** [9], a robotics dataset in which human-controlled robot arms move a variety of objects.

We compare EISEN to the (non-differentiable) affinity-based SSAP [13], the Transformer-based DETR [4], the centroid prediction-based Panoptic DeepLab (PDL) [6], and the region proposal-based Mask-RCNN [17]. All baselines require pixelwise segmentation supervision, for which we use the same motion signals as EISEN except for the conversion to pairwise connectivity. Because they were not designed to handle sparse supervision, we tune baseline object proposal hyperparameters to maximize recall. All models are evaluated on mIoU between ground truth and best-matched predicted segments; DETR and Mask-RCNN are not penalized for low precision.

4.2 Learning to segment from sparse object motion

EISEN outperforms standard architectures at motion-based learning.

Baseline segmentation architectures easily segment the **Playroom-val** set when given full supervision of all objects (Table 1, Full supervision.) When supervised only on RAFT-predicted optical flow, however, these models perform substantially worse (Table 1, Motion supervision) and exhibit characteristic qualitative failures (Figure 5), such as missing or lumping together objects.

EISEN, which treats object motion as an exclusively *relational* learning signal, performs well whether given full or motion-only supervision (Table 1.) Moreover, in contrast to the baselines, EISEN also accurately segments most objects in *test* scenes that differ from its training distribution in background, object number, and multi-object occlusion patterns (Figure 5; see Supplement.) These results suggest that only EISEN learns to detect the class of *Spelke objects* – the category-agnostic concept of “physical stuff that moves around together.” Interestingly, the strongest motion-supervised baseline is Mask-RCNN, which may

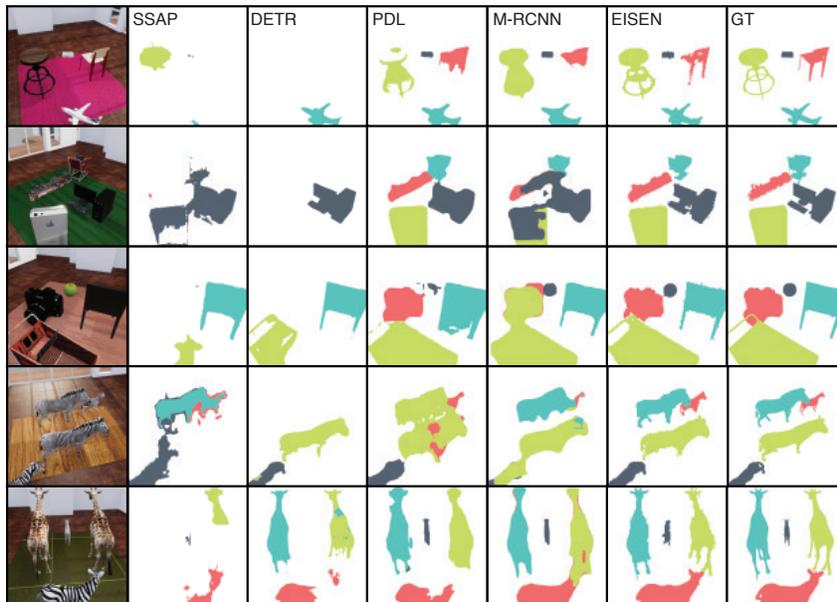


Fig. 5: **EISEN outperforms baselines at learning to segment from motion.** Segmentation predictions of EISEN and each baseline are shown for examples from the **Playroom** *val* set (top three rows) and *test* set (bottom two rows.) Baselines frequently lump, miss, and distort the shapes of objects. EISEN is able to capture fine details (e.g. the chair and giraffe legs) and segment closely spaced objects of similar appearance, (e.g. the zebras.)

implicitly use relational cues in its region proposal and non-maximal suppression modules to partly exclude false negative static regions of the scene.

4.3 Self-supervised segmentation of real-world scenes.

Learning to segment in the presence of background motion. The **Playroom** dataset has realistically complex Spelke objects but unrealistically simple motion. In particular, its scenes lack background motion and do not show the agentic mover of an object. Most video frames in the **DAVIS2016** dataset [27] have both object and (camera-induced) background motion, so we use it to test whether a useful segmentation learning signal can be extracted and used to train EISEN in this setting. Applying Competition to flow plateau maps often exposes a large background segment, which can be suppressed to yield a target object motion segment (Figure 6A; also see Supplement.) When this motion signal is used to train EISEN, the *static* segmentation performance on *held-out scenes* is 0.52, demonstrating that motion-based self-supervision supports learning of complex Spelke objects real scenes (Figure 6B.)

Unsupervised segmentation of the Bridge robotics dataset. We train EISEN for three rounds of bootstrapping to segment Spelke objects in **Bridge**

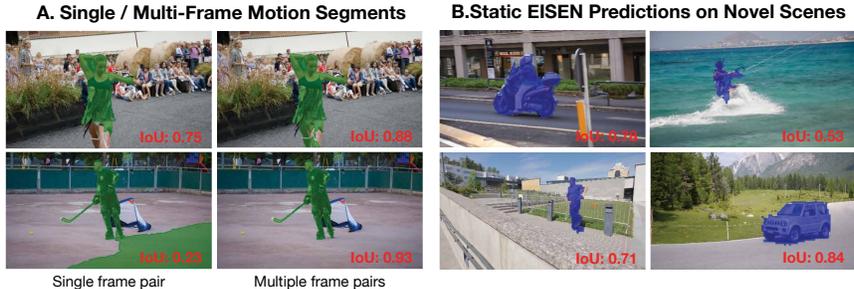


Fig. 6: EISEN learns to segment objects in *static, held-out scenes* on DAVIS2016. (A) Confident *teacher* segments computed from multiple frame pairs are better than those from a single frame pair. (B) Without any motion information, EISEN segments objects in *single RGB images* from held-out scenes.

Table 2: **Performance on Bridge after each round of bootstrapping.** EISEN improves at segmentation across three rounds by using its own inference pass to create better supervision signals. Neither Mask-RCNN nor Panoptic DeepLab perform well whether bootstrapped or pretrained on COCO.

Model	Round 1	Round 2	Round 3	Pretrained
MaskRCNN	0.053	0.081	0.102	0.070
Panoptic DeepLab	0.051	0.056	0.057	0.175
EISEN	0.336	0.453	0.551	-

(see Methods). EISEN’s segmentation of **Bridge** scenes dramatically improves with each round (Table 2 and Figure 7). In the first round, the model mainly learns to segment the robot arm, which is expected because this object is seen moving more than any other and the untrained EISEN teacher outputs few confident segments that could overwrite the raw motion training signal. In the subsequent rounds, top-down inference from the pretrained EISEN teacher modifies the raw motion signal; the improvement during these rounds suggests that top-down inference about physical scene structure can extract better learning signals than what is available from the raw image or motion alone. In contrast to EISEN, neither Mask-RCNN nor Panoptic DeepLab segment most of the objects either after applying the same bootstrapping procedure or when pretrained on COCO with categorical supervision (Table 2 and Figure 1.) EISEN’s combination of bottom-up grouping with top-down inference thus enables unsupervised segmentation of Spelke objects in real scenes.

4.4 Ablations of the EISEN architecture

Ablating KProp and Competition. EISEN performance on **Playroom** is nearly equal when using all affinity pairs versus using local and a small sample of long-range pairs (< 7% of total), though it drops slightly if long-range pairs

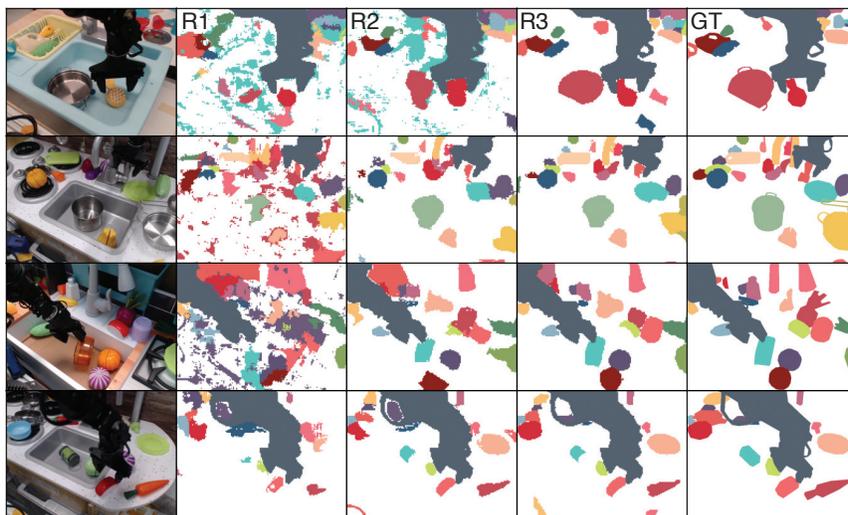


Fig. 7: **EISEN improves at segmenting Spelke objects with each round of bootstrapping.** After the first round of bootstrapping (R1), EISEN can segment the arm but few other objects well. Subsequent rounds (R2 and R3) substantially improve both the number of objects detected and their segmentation quality.

are omitted (Table 3). This suggests that plateau map alignment is mainly a local phenomenon and that grouping with EISEN relies heavily on local cues.

In contrast, the architectural components of KProp and Competition are essential for EISEN’s function. When either excitatory or inhibitory messages are ablated, or when using Softmax rather than ℓ^2 -normalization, performance drops nearly to zero (Table 3.) Moreover, the Competition module is better at extracting segments from the final plateau map than simply taking the Argmax over the channel dimension; this is expected, since the ℓ^2 -normalization in KProp does not encourage plateau map clusters to be one-hot vectors.

KProp and Competition are both RNNs, so their function may change with the number of iterations. Performance saturates only with > 30 KProp iterations and drops to near zero with a single iteration, implying that sustained message passing is essential: EISEN cannot operate as a feedforward model. In contrast, Competition requires only a single iteration on **Playroom** data (Table 3).

Ablating Affinity Prediction. Finally, we compare EISEN’s affinities to other affinity-like model representations. Object segments can be extracted from the attention maps of a self-supervised Vision Transformer (DINO [5]) using KProp and Competition, but their accuracy is well below EISEN’s; prior graph-based segmentation methods [31, 16, 11] do not detect **Playroom** objects as well as KProp-Competition (Table 4; see Supplement.) These experiments imply that EISEN is a better source of affinities than the (statically trained) DINO attention maps and that EISEN’s grouping network best makes use of both sources.

Table 3: **Ablations of EISEN.** Altering the architectural components of KProp or Competition drastically degrades performance, but only a small sample of long-range affinities are necessary (Left). Lowering the number of RNN iterations for KProp or Comp gradually degrades performance (Right).

messages	affinity	norm	readout	mIoU	KProp iters	Comp iters	mIoU
Ex+Inb	Loc+Glob	ℓ^2	Comp	0.730	40	3	0.730
Ex+Inb	Loc	ℓ^2	Comp	0.700	30	3	0.720
Ex+Inb	Full	ℓ^2	Comp	0.732	20	3	0.697
Ex+Inb	Loc+Glob	ℓ^2	Argmax	0.676	10	3	0.389
Ex	Loc+Glob	ℓ^2	Comp	0.036	1	3	0.052
Inb	Loc+Glob	ℓ^2	Comp	0.036	40	2	0.730
Ex+Inb	Loc+Glob	softmax	Comp	0.036	40	1	0.729

Table 4: **Comparison of DINO and EISEN affinities with different graph clustering algorithms.** EISEN affinities are downsampled to the same size as DINO affinities for a fair comparison

Model	Spectral clustering	LabelProp	AffinityProp	KProp+comp
DINO	0.354	0.135	0.255	0.545
EISEN	0.062	0.084	0.319	0.684

5 Conclusion

We have proposed EISEN, a fully differentiable, graph-based grouping architecture for learning to segment Spelke objects. While our algorithm performs on par with prior segmentation models when fully supervised (Table 1), its main strength is an ability to learn *without supervision*: by applying top-down inference with its own segmentation predictions, it progressively improves motion-based training signals. These key architecture and learning innovations are critical for dealing with the challenges of unsupervised, category-agnostic object segmentation in real-world scenes (Figure 2.) Since EISEN is based on the principle of grouping things that move together, it cannot necessarily address higher-level notions of “objectness” that include things rarely seen moving (e.g., houses and street signs.) It will therefore be important in future work to explore the relationship between motion-based and motion-independent object learning and identify deeper principles of grouping that extend to both.

Acknowledgements J.B.T is supported by NSF Science Technology Center Award CCF-1231216. D.L.K.Y is supported by the NSF (RI 1703161 and CAREER Award 1844724) and hardware donations from the NVIDIA Corporation. J.B.T. and D.L.K.Y. are supported by the DARPA Machine Common Sense program. J.W. is in part supported by Stanford HAI, Samsung, ADI, Salesforce, Bosch, and Meta. D.M.B. is supported by a Wu Tsai Interdisciplinary Scholarship and is a Biogen Fellow of the Life Sciences Research Foundation. We thank Chaofei Fan and Drew Linsley for early discussions about EISEN.

References

1. Arora, T., Li, L.E., Cai, M.B.: Learning to perceive objects by prediction. In: SVRHM 2021 Workshop@ NeurIPS (2021)
2. Bear, D., Fan, C., Mrowca, D., Li, Y., Alter, S., Nayebi, A., Schwartz, J., Fei-Fei, L.F., Wu, J., Tenenbaum, J., et al.: Learning physical graph representations from visual scenes. *Advances in Neural Information Processing Systems* **33**, 6027–6039 (2020)
3. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: A. Fitzgibbon et al. (Eds.) (ed.) *European Conf. on Computer Vision (ECCV)*. pp. 611–625. Part IV, LNCS 7577, Springer-Verlag (Oct 2012)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: *European conference on computer vision*. pp. 213–229. Springer (2020)
5. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 9650–9660 (2021)
6. Cheng, B., Collins, M.D., Zhu, Y., Liu, T., Huang, T.S., Adam, H., Chen, L.C.: Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 12475–12485 (2020)
7. Dorfman, N., Harari, D., Ullman, S.: Learning to perceive coherent objects. In: *Proceedings of the Annual Meeting of the Cognitive Science Society*. vol. 35 (2013)
8. Du, Y., Smith, K., Ulman, T., Tenenbaum, J., Wu, J.: Unsupervised discovery of 3d physical objects from video. *arXiv preprint arXiv:2007.12348* (2020)
9. Ebert, F., Yang, Y., Schmeckpeper, K., Bucher, B., Georgakis, G., Daniilidis, K., Finn, C., Levine, S.: Bridge data: Boosting generalization of robotic skills with cross-domain datasets. *arXiv preprint arXiv:2109.13396* (2021)
10. Follmann, P., Bottger, T., Hartinger, P., König, R., Ulrich, M.: Mvtec d2s: densely segmented supermarket dataset. In: *Proceedings of the European conference on computer vision (ECCV)*. pp. 569–585 (2018)
11. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. *science* **315**(5814), 972–976 (2007)
12. Gan, C., Schwartz, J., Alter, S., Schrimpf, M., Traer, J., De Freitas, J., Kubilius, J., Bhandwaldar, A., Haber, N., Sano, M., et al.: Threedworld: A platform for interactive multi-modal physical simulation. *arXiv preprint arXiv:2007.04954* (2020)
13. Gao, N., Shan, Y., Wang, Y., Zhao, X., Yu, Y., Yang, M., Huang, K.: Ssap: Single-shot instance segmentation with affinity pyramid. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 642–651 (2019)
14. Girshick, R.: Fast r-cnn. In: *Proceedings of the IEEE international conference on computer vision*. pp. 1440–1448 (2015)
15. Greff, K., Kaufman, R.L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M., Lerchner, A.: Multi-object representation learning with iterative variational inference. In: *International Conference on Machine Learning*. pp. 2424–2433. PMLR (2019)
16. Gregory, S.: Finding overlapping communities in networks by label propagation. *New journal of Physics* **12**(10), 103018 (2010)

17. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
18. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
19. Hinton, G.: How to represent part-whole hierarchies in a neural network. arXiv preprint arXiv:2102.12627 (2021)
20. Kabra, R., Zoran, D., Erdogan, G., Matthey, L., Creswell, A., Botvinick, M., Lerchner, A., Burgess, C.: Simone: View-invariant, temporally-abstracted object representations via unsupervised video decomposition. *Advances in Neural Information Processing Systems* **34** (2021)
21. Kipf, T., Elsayed, G.F., Mahendran, A., Stone, A., Sabour, S., Heigold, G., Jonschkowski, R., Dosovitskiy, A., Greff, K.: Conditional object-centric learning from video. arXiv preprint arXiv:2111.12594 (2021)
22. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
23. Liu, W., Rabinovich, A., Berg, A.C.: Parsenet: Looking wider to see better. arXiv preprint arXiv:1506.04579 (2015)
24. Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., Kipf, T.: Object-centric learning with slot attention. *Advances in Neural Information Processing Systems* **33**, 11525–11538 (2020)
25. Luo, L., Xiong, Y., Liu, Y., Sun, X.: Adaptive gradient methods with dynamic bound of learning rate. arXiv preprint arXiv:1902.09843 (2019)
26. Peng, B., Zhang, L., Zhang, D.: A survey of graph theoretical approaches to image segmentation. *Pattern recognition* **46**(3), 1020–1038 (2013)
27. Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 724–732 (2016)
28. Roelfsema, P.R., et al.: Cortical algorithms for perceptual grouping. *Annual review of neuroscience* **29**(1), 203–227 (2006)
29. Ross, M.G., Kaelbling, L.P.: Segmentation according to natural examples: learning static segmentation from motion segmentation. *IEEE transactions on pattern analysis and machine intelligence* **31**(4), 661–676 (2008)
30. Sabour, S., Tagliasacchi, A., Yazdani, S., Hinton, G., Fleet, D.J.: Unsupervised part representation by flow capsules. In: International Conference on Machine Learning. pp. 9213–9223. PMLR (2021)
31. Shi, J., Malik, J.: Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence* **22**(8), 888–905 (2000)
32. Siméoni, O., Puy, G., Vo, H.V., Roburin, S., Gidaris, S., Bursuc, A., Pérez, P., Marlet, R., Ponce, J.: Localizing objects with self-supervised transformers and no labels. arXiv preprint arXiv:2109.14279 (2021)
33. Spelke, E.S.: Principles of object perception. *Cognitive science* **14**(1), 29–56 (1990)
34. Tangemann, M., Schneider, S., von Kügelgen, J., Locatello, F., Gehler, P., Brox, T., Kümmerer, M., Bethge, M., Schölkopf, B.: Unsupervised object learning via common fate. arXiv preprint arXiv:2110.06562 (2021)
35. Teed, Z., Deng, J.: Raft: Recurrent all-pairs field transforms for optical flow. In: European conference on computer vision. pp. 402–419. Springer (2020)
36. Todorovic, D.: Gestalt principles. *Scholarpedia* **3**(12), 5345 (2008)

37. Tsao, T., Tsao, D.Y.: A topological solution to object segmentation and tracking. arXiv preprint arXiv:2107.02036 (2021)
38. Ullman, S., Harari, D., Dorfman, N.: From simple innate biases to complex visual concepts. *Proceedings of the National Academy of Sciences* **109**(44), 18215–18220 (2012)
39. Wang, Y., Shen, X., Hu, S., Yuan, Y., Crowley, J., Vafreydaz, D.: Self-supervised transformers for unsupervised object discovery using normalized cut. arXiv preprint arXiv:2202.11539 (2022)
40. Yang, C., Lamdouar, H., Lu, E., Zisserman, A., Xie, W.: Self-supervised video object segmentation by motion grouping. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 7177–7188 (2021)
41. Zhou, H., Friedman, H.S., Von Der Heydt, R.: Coding of border ownership in monkey visual cortex. *Journal of Neuroscience* **20**(17), 6594–6611 (2000)
42. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159 (2020)