Discovering Deformable Keypoint Pyramids

Jianing Qian, Anastasios Panagopoulos, and Dinesh Jayaraman

University of Pennsylvania
{jianingq,anpans,dineshj}@seas.upenn.edu

Abstract. The locations of objects and their associated landmark keypoints can serve as versatile and semantically meaningful image representations. In natural scenes, these keypoints are often hierarchically grouped into sets corresponding to coherently moving objects and their moveable and deformable parts. Motivated by this observation, we propose Keypoint Pyramids, an approach to exploit this property for discovering keypoints without explicit supervision. Keypoint Pyramids discovers multi-level keypoint hierarchies satisfying three desiderata: comprehensiveness of the overall keypoint representation, coarse-to-fine informativeness of individual hierarchy levels, and parent-child associations of keypoints across levels. On human pose and tabletop multi-object scenes, our experimental results show that Keypoint Pyramids jointly discovers object keypoints and their natural hierarchical groupings, with finer levels adding detail to coarser levels to more comprehensively represent the visual scene. Further, we show qualitatively and quantitatively that keypoints discovered by Keypoint Pyramids using its hierarchical prior bind more consistently, and are more predictive of manually annotated semantic keypoints, compared to prior flat keypoint discovery approaches. Code is at: https://github.com/jianingq/KeypointPyramids

Keywords: keypoint, self-supervision, hierarchical representations

1 Introduction

Object keypoint sets are particularly attractive in computer vision as compact and versatile representations of images. In common instantiations of this idea, each keypoint in an image is represented by pixel coordinates attached to a specific semantic object in the real scene, and usually to a specific landmark 3D position on its surface. When all such keypoints in a scene are combined, the resulting scene descriptor is succinct, easy to interpret semantically, and convenient for spatial reasoning and systematic generalization. These advantages have been explored by researchers over many years for a large number of applications spanning pose estimation for humans [3], animals [33], and objects [39], face recognition [34], tactile sensing [26], reinforcement learning in video games [24, 35] , and robotics [2, 4, 32, 40].

Early applications of keypoints [8, 10, 34] relied on pre-annotated fiducial keypoints for select object categories, such as the joints of a human skeleton. However, recent works [2, 13, 19, 24, 36, 44, 49] have targeted *discovering* object

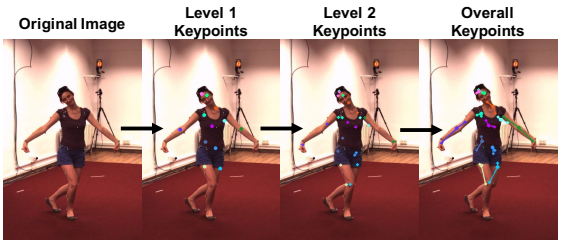

Fig. 1. Our method, Keypoint Pyramids, discovers multi-level keypoint hierarchies without explicit supervision, and represent information in a coarse-to-fine structure to represent the configurations of objects and their moving and deformable parts and subparts.

keypoint representations without such explicit supervision to extend the benefits of keypoint representations beyond only a few pre-annotated object categories. These approaches build off the recent successes of general unsupervised image representation learning that produce unstructured 2D feature maps or 1D vector representations of images. To inject keypoint structure into such representations, unsupervised keypoint discovery methods rely on two fundamental properties of object keypoints: sparsity and local associations with small neighborhoods in the image. This prior knowledge about keypoints is commonly represented through a representational bottleneck [5] that enforces sparse and local keypoints.

In this paper, we start by observing an additional, higher order property of keypoint sets: keypoints in natural scenes are often hierarchically grouped into nested subsets that are tied to coherently moving objects and their movable and deformable parts. In a multi-object scene, each object may coarsely be represented by a single keypoint to specify the location of that object as a whole. To capture more fine-grained detail such as its pose, each rigid object requires two additional keypoints (three in total) to specify its 6-degree-of-freedom pose. An articulated object containing multiple parts requires more keypoints for each part, and a continuously deformable object may be modeled as containing many local neighborhoods each containing many keypoints.

Motivated by this natural hierarchical organization, we argue for representing the configuration of objects in a scene in a hierarchical data structure containing nested groups of keypoints. We propose Keypoint Pyramids, an unsupervised approach that learns to represent images as coarse-to-fine keypoint hierarchies, improving upon current approaches that discover *flat* keypoint representations. Keypoints in the earlier coarser levels of this hierarchy capture only the gist of the scene. Later finer levels can then add new, more local keypoints to elaborate upon this and describe the scene more comprehensively. Each l-level keypoint in the pyramid is connected through spring connections to several children keypoints at its subsequent finer level $l + 1$. For example, a human may be represented at the coarsest level by a single keypoint to identify their location in a scene. In the next level, important joints determining overall body pose such as the shoulders, elbows, and knees may be represented. At subsequent levels, finer details such

as the fingers on the hand, and facial keypoints determining facial expressions may be modeled.

Indeed, several prior works have established the utility of manually defined hierarchies over pre-annotated keypoints [11, 17, 22, 37]. Our approach, Keypoint Pyramids, is the first to exploit this for *unsupervised* keypoint discovery, improving the quality of discovered keypoint representations, and providing a convenient coarse-to-fine representation for downstream use cases. Through quantitative and qualitative evaluations on several datasets of human and multi-object images, we establish that learned keypoint pyramids generate better descriptions of visual scenes, showing higher quality information retention and more consistent keypoint binding than prior approaches that all generate flat keypoint sets. Our results validate hierarchical organization as an important prior for keypoint discovery, and our Keypoint Pyramids approach as an effective technique to exploit this prior.

2 Related Work

Unsupervised object-centric representations: Explicitly representing objects within the feature representation has many benefits, including improved ML generalization to novel compositions of similar objects [15, 25]. To extend such benefits of object-centric representations and reasoning beyond just the tens of categories for which pre-annotated bounding boxes and segmentation masks exist, many recent works have aimed to discover self-supervised object-centric representations without any manual annotations. One class of such methods aims to partition the scene into object bounding boxes or segmentation masks [1, 6, 14, 21, 23, 29, 30, 45, 50]. To represent pose and other variations internal to the bounding boxes or contours of each discovered object, these methods rely on unstructured dense feature vectors. Instead, we aim to comprehensively represent the full object configuration of a multi-object visual scene through a versatile, sparse, and succinct collection of keypoints that can not only localize objects but also capture their pose, part articulations and deformations.

There is also work on object part discovery [38, 42, 43, 47] by grouping local features into semantically consistent parts. For example, it is possible to exploit optical flow information for part discovery [47], or discover 3D shape primitives for a target mesh [38], or to spatially downsample object-based feature maps to represent hierarchies in simulated and simple scenes [43]. These methods all operate in simplified settings [38, 43, 47], rely on additional information for part discovery [42, 47], and/or inherit dense feature vector-based representations of parts [42, 43]. We avoid these pitfalls in our Keypoint Pyramids approach.

Unsupervised keypoint discovery: More relevant to us, unsupervised keypoint discovery methods represent an input image as a set of landmarks that describe the object configuration [2, 13, 19, 24, 36, 44, 49]. Zhang et al. [49] design an hourglass network that takes in a single image and outputs a set of landmarks that describe the object shapes. Lorenz et al. [31] designs part discovery network that aims to disentangled object shape and appearances. Thewlis

et al. [44] constrain learned landmarks for an object category to be viewpoint-invariant. Jakab et al. [19] carefully design a network architecture that is now called KeyNet, which uses extracted keypoints as an information bottleneck [5] to reconstruct the input image. Since their work, many others have built on KeyNet, for example, using a spatial feature prediction error map as input to KeyNet [13], inputting additional pose prior images to KeyNet [20], or training the keypoint outputs to be predictive of future frames [35]. Kulkarni et al. [24] augment KeyNet with a feature map tied closely to keypoints by constructing a "transported feature map" bottleneck. We build upon these approaches, particularly the keypoint and transported feature map representations of [19] and [24], but unlike any prior approaches, we jointly discover not only a flat set of keypoints, but also their hierarchical organization, with different levels corresponding to objects and their articulated and deformable parts and subparts in a coarse-to-fine structure. As described in Section 1, prior approaches only exploit the sparsity and local association properties of individual keypoints, but we observe and exploit the joint hierarchical organization property of keypoint sets in an image. In our experiments, we compare against flat keypoint discovery methods and show superior results.

Supervised hierarchies: Many previous works have shown benefits from modeling manually annotated hierarchies over keypoints or objects. The classic pictorial structures model [7, 9, 10] established the utility of predefined graphs over object parts for object recognition. For detecting supervised keypoints, Huang et al. [16] propose a coarse-to-fine training and detection process, yielding advantages across human and bird pose detection tasks. In this case, "coarse" detections are merely less accurate detections of *all* keypoints, to be refined afterwards. Samet and Akbas [41] use a predefined hierarchy of over 133 annotated human body keypoints to perform "hierarchical regression", proceeding in stages to regress finer keypoints such as facial features, conditioned on coarser keypoints that determine the body pose. Mrowca et al. [37] show how predefined dense particle-based models of rigid or deformable objects can be abstracted into clustered hierarchies to allow efficiently modeling complex physical dynamics such as non-rigid collisions from data. Broadly, these methods showcase the utility of keypoint hierarchies, but our method is different in that it operates without supervision for either the keypoints or their hierarchical relationships, and aims to discover both jointly from images alone.

3 Approach

Suppose we are given a dataset of unlabeled images from a domain, such as humans in varied poses, or rooms with various configurations of objects. With no prior annotations, can we automatically learn to succinctly represent new images from that domain in terms of their objects, object parts, and other useful landmarks?

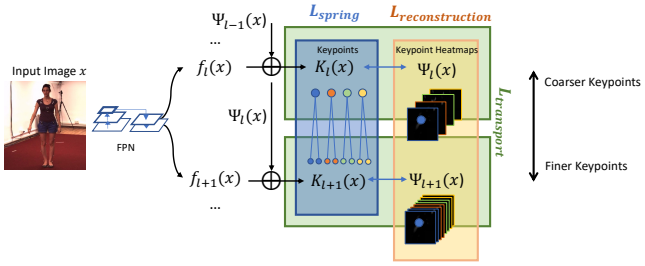

Fig. 2. The Keypoint Pyramids encoder generates a coarse to fine hierarchy of keypoints for an input image. This schematic illustrates two consecutive levels $K_l(x)$ and $K_{l+1}(x)$. Details in Sec 3.1. The three training losses annotated on the right: "combined", "transport", and "spring" are described in detail in Section 3.2, 3.3, and 3.4.

Three desiderata for keypoint hierarchies. To accomplish this, as motivated in Section 1, we would like to represent an image as a keypoint hierarchy that satisfies three desiderata:

1. It must permit comprehensively describing the configuration of the objects in the image i.e., their locations, poses, articulations, and deformations.
2. Individual levels in the hierarchy must proceed from coarse to fine, representing different trade-offs between compactness and informativeness.
3. Each keypoint at a coarse level must be tied to a set of "children" keypoints at the next finer level, which help provide more details about their parent keypoint.

We propose Keypoint Pyramids, an approach to learn representations that satisfy these desiderata from datasets of unlabeled images. Fig 2 illustrates the overall workflow of our method. In Section 3.1, we describe the Keypoint Pyramids neural architecture for encoding input images into keypoint hierarchies. Next, we describe how to train Keypoint Pyramids through a training objective that balances the three desiderata above, laid out in Sections 3.2, 3.3, and 3.4. In addition to the input image and the encoding weights above, our training procedure for the encoder relies on auxiliary reference images for each training sample, and auxiliary network weights that aid in training. Finally, we summarize the overall objective and describe implementation and optimization details in Section 3.5.

3.1 Keypoint Pyramids encoder architecture

First, we define a neural network architecture for encoding images $x \in \mathbb{R}^{H \times W \times 3}$ to L-level keypoint hierarchies. Figure 2 shows a schematic. At each level $1 \leq l \leq L$, we wish to generate a new set of N_l keypoints $K_l(x) = [k_l^1(x), ..., k_l^{N_l}(x)]$. Each keypoint $k_l^n(x)$ is a 2-D vector representing pixel coordinates within the image.

To generate such keypoint hierarchies, we use a feature pyramid network (FPN) architecture [28] to extract feature maps at L scales, denoted as $\{f_l(x)\}_{l=1}^{L}$, matched to the L levels of the keypoint hierarchy. At the first level $l = 1$, we generate a keypoint set $K_1(x)$ from the coarsest, smallest scale feature maps $f_1(x)$ through a convolutional keypoint encoder network.

At subsequent finer layers $l > 1$, we condition keypoint encoders additionally on $K_{l-1}(x)$, so that lower levels in the keypoint pyramid can be influenced by higher levels. Specifically, we transform k_{l-1}^n, the n-th keypoint coordinates at level $l-1$, to a heatmap representation $\Psi_{l-1}^n(x) \in \mathbb{R}^{H \times W}$ by applying a Gaussian function with a small fixed variance around the keypoint coordinates. Note that the heatmaps $\Psi_{l-1}^n(x)$ are lossless representations of the keypoint coordinates $k_{l-1}^n(x)$ and we will go back and forth between these two representations as convenient. The stack of all N_{l-1} heatmaps at level $l-1$ is denoted Ψ_{l-1}. The inputs to the keypoint encoder for generating the l-th level $K_l(x)$ are then $[\Psi_{l-1}(x), f_l(x)]$. This architecture is shown in Figure 2.

Keypoint encoders at each level l follow the popular KeyNet architecture [19]: a convolutional network takes $f_l(x)$ as input and generates N_l feature maps, to each of which a spatial softmax operation is applied followed by marginalization along the image dimensions to determine the keypoint coordinates $k_l^n(x)$.

3.2 Comprehensiveness of the overall keypoint representation

Our first desideratum for the keypoint hierarchy is that it should permit a comprehensive description of the object configuration in the scene through keypoint coordinates alone. To achieve this, our objective includes a loss term that measures the pixelwise error for reconstructing the input image x from the combination of all levels of the keypoint hierarchy. The process of calculating this objective is shown in Figure 3.

First, we convert all keypoints $k_l^n(x)$ across all levels to their corresponding heatmaps $\Psi_l^n(x)$ as described in Section 3.1. Stacking these heatmaps across all levels, we get $\Psi(x) \in \mathbb{R}^{H \times W \times \sum_l N_l}$. This keypoint heatmap stack $\Psi(x)$ is now fed into a decoder that is to be trained to reconstruct the input image x. However, keypoints only capture object configurations, and do not contain information about other aspects of the appearance of the scene such as the background, lighting, and colors. To provide this auxiliary information required for image reconstruction, following Jakab et al. [19], we extract convolutional "appearance" features $\Phi(x_{\text{ref}}) \in \mathbb{R}^{H \times W \times C}$ from a reference image x_{ref} of the same scene as x, but with a different configuration of the objects. For example, x_{ref} could be a different video frame from the same static-camera video sequence as x.

Finally, we train a convolutional decoder network to map from the concatenation $[\Psi(x), \Phi(x_{\text{ref}})]$ to a reconstruction $\hat{x}$, minimizing the following combined reconstruction objective:

$$\mathcal{L}_{\text{reconstruction}}(\widehat{x}, x) = ||\widehat{x}([\Psi(x), \Phi(x_{\text{ref}})]) - x||_2^2. \tag{1}$$

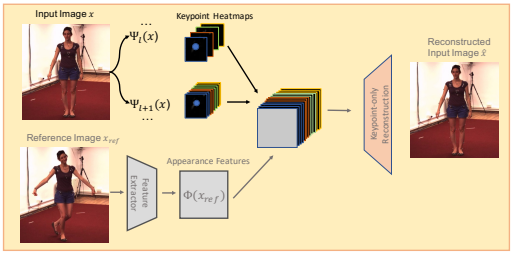

Fig. 3. To train a comprehensive keypoint representation, we reconstruct the input image from the combination of all levels of our hierarchy, generating a reconstruction loss. Details in Section 3.2. The gray areas in the figure show components that are required only to compute the training objectives; these are not used at test time.

3.3 Graded informativeness of keypoint levels

To satisfy our second desideratum, individual levels in the keypoint hierarchy must each capture useful information, and finer levels must progressively capture more information. However, the combined reconstruction objective of Equation 1 pools keypoints from all levels in the hierarchy and does not impose any requirements on individual levels. For example, it would suffice to minimize Equation 1 if all of the information was represented in only one level, and all other levels captured no information at all.

Augmenting keypoint coordinates with local features. To incentivize meaningful coarse-to-fine keypoint hierarchies, we introduce level-wise objective terms requiring each level to be independently informative about the object configuration. However, note that merely the 2D pixel coordinates of keypoints at coarse levels cannot capture the fine-grained details of the object configuration. For example, just the coordinates of the centroid of a person cannot reasonably be sufficient to infer their full pose. To effectively capture the object configuration, keypoint coordinates at each level must therefore be augmented with some residual information from their local image neighborhoods, to substitute for missing finer-level keypoints.

For this purpose, we construct feature-augmented keypoints. Specifically, to represent missing fine-grained information from level l keypoints, we extract convolutional features from their neighborhoods. We compute feature maps $\Phi_l(x) \in \mathbb{R}^{H \times W \times C_l}$ from a new convolutional encoder operating on top of the FPN level l features $f_l(x)$. Then, local features around a keypoint $k_l^n(x)$ can be computed by masking these features through an elementwise product with the keypoint heatmap $\Psi_l^n(x)$. This produces the feature-augmented keypoints $[\Psi_l^n(x), \Phi_l(x)\Psi_l^n(x)]$.

Finally, to incentivize finer levels to capture more information within the keypoint coordinates, we augment the heatmaps for coarser levels with more information than for finer levels; accordingly, we set the number of channels C_l

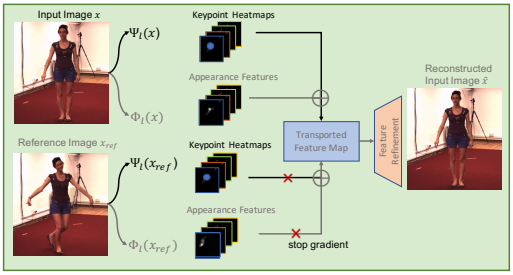

Fig. 4. While individual levels of the keypoint hierarchy need not be comprehensive, they should each capture useful information. We construct feature-augmented keypoints at each level, create "transported" feature maps and then use those to reconstruct the image, generating a transport loss. Details in Section 3.3.

in the feature maps $\Phi_l(x)$ to be higher for coarser levels, i.e., $C_{l_1} > C_{l_2}$ for $l_1 < l_2$.

Transport loss. Recall that the combined reconstruction objective Equation 1 aims to reconstruct the input image x using appearance features from a reference image x_{ref} and keypoints from the original input x. In similar spirit, we may now set up a level-wise image reconstruction objective using the feature-augmented keypoints above. In other words, we would like to compute appearance features $\Phi_l(x_{\text{ref}})$ from the reference image, inject augmented keypoint information from x, and train a decoder to produce a reconstruction $\hat{x}$. See Figure 4.

We set up such layerwise reconstruction losses following the "keypoint transport" loss from Kulkarni et al. [24]. At each level, we first compute a transported feature map:

$$\Phi'_l(x, x_{\text{ref}}) = \underbrace{\Phi_l(x_{\text{ref}})(1 - \Psi_l(x_{\text{ref}}))(1 - \Psi_l(x))}_{\text{Appearance features from reference image}} + \underbrace{\Phi_l(x)(1 - (1 - \Psi_l(x_{\text{ref}}))(1 - \Psi_l(x)))}_{\text{"Augmented" keypoints from input image}}. \quad (2)$$

This transport equation can be interpreted as follows. The first term effectively removes all keypoints from the reference image feature map to provide reference appearance information alone. The second term fills in those keypoint holes using augmented keypoints from the original image x. We can now reconstruct x from this transported feature at each level l, computing a transport objective:

$$\mathcal{L}_{\text{transport}} = \sum_{l=1}^{L} \lambda_l ||\widehat{x}_l(\Phi'_l(x, x_{\text{ref}})) - x||_2^2. \quad (3)$$

3.4 Keypoint associations across levels

Finally, how can we ensure a pyramidal association structure between higher and lower levels, as in our third desideratum? We pre-specify desired associations

between keypoints across neighboring levels and use a spring loss to encourage children keypoints at finer levels to remain close to their parent.

For each "parent" keypoint $k_l^n \in K_l$ at level l, we specify a fixed disjoint subset $\delta_l^n \subset K_{l+1}$ of children keypoints at its finer level. In our experiments, we use $\delta_l^n = \{Ml + 1, Ml + 2, \ldots, M(l + 1) - 1\}$, where M specifies the number of children per parent keypoint. We would now like each parent keypoint k_l^n to serve as an anchor for its children. We therefore penalize the deviation between children keypoint coordinates and their parent. This is akin to minimizing the energy of a mechanical system of springs connecting each child to its parent. This produces the following spring loss:

$$\mathcal{L}_{\text{spring}} = \lambda_s \sum_{\text{levels } l<L} \sum_{\text{keypoints } n \leq N_l} \sum_{\text{children } m \in \delta_l^n} ||k_l^n - k_{l+1}^m||_2^2 \tag{4}$$

3.5 Implementation details

The overall Keypoint Pyramids objective function is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{reconstruction}} + \mathcal{L}_{\text{transport}} + \mathcal{L}_{\text{spring}} \tag{5}$$

We minimize this objective end-to-end, jointly training the feature pyramid network, the keypoint encoders for all levels, as well as the auxiliary weights required during training, namely, the feature extractors and decoders. We use Adam optimizer with learning rate of 1e-4 for all experiments. During training, we randomly sampled a reference image x_{ref} from the same video sequence as the input image x, within 250 frames from it. In all of our experiments, we train Keypoint Pyramids with $L = 2$ levels, and with $N_1 = 10$ and $N_2 = 20$ keypoints on the first and second levels. Thus, our combined flattened representation has $\sum_l N_l = 30$ keypoints. We set $\lambda_s = 1, \lambda_1 = 0.1$ and $\lambda_2 = 1$ for all of our experiments. For the FPN network, we output feature maps at two levels and $l = 2$. At the coarsest level, the feature maps have size 16×16 and at the finest level the feature maps have size 32×32.

4 Experiments

Our experiments aim to answer the following questions: (1) Does Keypoint Pyramids discover semantically meaningful keypoint hierarchies? (2) Compared to prior flat keypoint discovery approaches, how well does a flattened Keypoint Pyramid recover the configurations of objects in the scene? (3) How important are the different components of our approach? (4)Is the Keypoint Pyramid representation suitable for downstream computer vision tasks?

4.1 Datasets

While many prior unsupervised keypoint and object discovery approaches have been evaluated in simulated settings, we focus on two real image datasets to

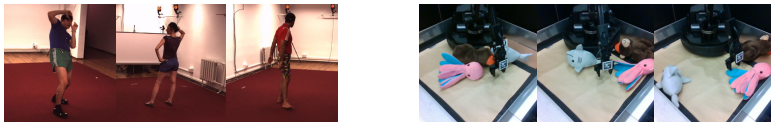

Fig. 5. Sample images from the two datasets used in our experiments: **(left)** Human 3.6M (H3.6M) showing people enacting various actions, and **(right)** our new dataset Bot-and-Objects (B&O), containing a robot interacting with objects on a tabletop.

evaluate Keypoint Pyramids on realistic scenes and objects.

Human3.6M (H3.6M): Human3.6M [18] is a large-scale video dataset featuring 7 actors performing 16 categories of actions in an indoor environment. It contains 3.6M images. Following the conventions in [27], we use 5 human subjects (S1, S5, S6, S7, S8) for training and the remaining 2 human subjects (S9, S11) for testing. Image pairs (x, x_{ref}) are extracted from the same video sequence. We apply loose crops around the subject using ground-truth annotation following [19]. To focus on the full body pose, we omit 5 action categories (Sitting, Smoking, etc.) that involve largely seated poses, leaving 11 categories in our dataset. This dataset is challenging because it requires the network to learn to recognize common keypoints that generalize across actors with disparate appearances, clothing, and body shapes, set against different backgrounds[1] and non-ideal lighting conditions. Further, modeling the human body is challenging, because it is a complicated articulated and deformable object with many moving parts and other degrees of freedom. On the other hand, this dataset permits extensive quantitative evaluation: it contains exhaustive keypoint annotations for 17 human pose keypoints corresponding to the major joints for all images, and the action category labels also permit an action recognition task from discovered keypoint representations.

Bot-And-Objects (B&O): To evaluate Keypoint Pyramids on real-world multi-object scenes, we collect an object pushing dataset with an articulated 5-degree-of-freedom WidowX 200 robot arm and three plush toys. We collect a video dataset with 450 videos, each containing 30 frames (13500 images). Between any two frames, the robot arm performs random motions of its gripper up to 5 cm within its 50 by 50 cm workspace, frequently displacing or rotating objects, and thus generating diverse object configurations within our dataset. We train on 10800 images and test on the remaining 2700 images.

4.2 Baselines and Ablations

Recall that all prior keypoint discovery approaches produce flat keypoint sets. We pick two state-of-the-art approaches for comparison against Keypoint Pyramids:

- **KeyNet** [19]: This baseline uses a flat keypoint set output by an encoder as the bottleneck in a neural network autoencoder.

[1] different video sequences are shot against different backgrounds

- **Transporter** [24]: This method trains convolutional feature maps alongside keypoints to permit reconstruction from a transported feature map.

These baselines are the most widely used object keypoint discovery method so far. [13, 20, 35] mentioned above all reuse the KeyNet encoder architecture, and many works reuse the Transporter loss [46, 48]. A comparison with a more recent method [12] is in Sec A.3. For both baselines, we use the widely used KeyNet-based keypoint encoder network architecture (same as for our method), with inputs from the largest and final feature map from FPN, which has size 32×32. More architecture details are in Sec A.1. We train both baselines with varying numbers of keypoints for fair comparison against our Keypoint Pyramids approach. In addition to these baselines, we also evaluate several ablations of our approach to analyze the effects of its various components. First, we train without the combined reconstruction loss (**No-Reconstruction**), without the transport loss (**No-Transport**), and without the spring loss (**No-Spring**). Next, rather than use the keypoints-only reconstruction loss for the overall flattened representation (Equation 1) and the augmented keypoints-based transport loss for the individual levels (Equation 3), we try using the same type of loss for both, either keypoint-only reconstruction (**All-Reconstruction**) or transport loss (**All-Transport**). We also run an ablation without the architectural choice of conditioning the keypoint encoder at level l on the keypoints from the previous level $l-1$ (**Unconditioned**). Note that Unconditioned still introduces dependencies between keypoint levels during training, through the reconstruction and spring losses. Finally, for our full method after training, we evaluate keypoints from its individual levels separately (**Level l=1 or 2**) to validate the coarse-to-fine representation.

4.3 Results

On H3.6M, which comes with exhaustive annotations for 17 major joints, we report the RMSE error for linear regression from discovered keypoints to annotated ground truth keypoint coordinates. This quantitatively evaluates keypoint representations for their ability to capture object configurations. We split the test data into two halves, fit the regression on one half and report errors on the other half. Table 1 shows the keypoint regression error for all methods for levels 1, 2, and for the combined flattened keypoint representation (level 1 + level 2). For comparison with flat approaches, we train them three times with 10, 20, and 30 keypoints. Flattened Keypoint Pyramids performs much better than intrinsically flat approaches, and shows a clear progression from level 1 to level 2 to the flattened representation. Our ablations further validate our algorithm design choices. All-Reconstruction which uses only keypoint information from source images works better than All-Transport which always augments keypoints with local feature information, but neither works as well as our choice to combine the overall reconstruction loss with the level-wise transport loss to allow graded information in the individual levels. Further, the No-X ablations show that all terms in the objective function are important to our performance: dropping any

	Methods ↓ / Level (num. keypts.) →	level 1(10)	level 2(20)	flattened(30)
	Transporter	50.15	45.25	47.45
	KeyNet	56.51	53.28	46.71
	Keypoint Pyramids (Ours)	52.81	**43.97**	**43.30**
Ablations	(KP) No-Reconstruction	**49.09**	46.73	45.23
	(KP) No-Transport	50.52	46.29	45.85
	(KP) No-Spring	49.27	48.21	45.28
	(KP) All-Transport	54.72	50.07	49.53
	(KP) All-Reconstruction	50.83	48.15	46.93
	(KP) Unconditioned	49.37	44.73	**43.75**

Table 1. Keypoint regression error on H3.6M, compared to prior flat unsupervised keypoint discovery baselines and ablations. Lower is better.

Methods	Accuracy
Ground-truth Keypoints(17)	0.331
Keypoint Pyramids (Ours)(L1+L2)	**0.218**
Transporter(30)	0.177
KeyNet(30)	0.179
Keypoint Pyramids (Ours)(L2)	**0.193**
Transporter(20)	0.164
KeyNet(20)	0.168
Keypoint Pyramids (Ours)(L1)	**0.182**
Transporter(10)	0.152
KeyNet(10)	0.148

Table 2. Action classification accuracy on H3.6M, compared to prior flat unsupervised keypoint discovery baselines. Higher is better.

individual term deteriorates performance and all terms contribute nearly equally. Finally, removing forward connections from coarser to finer keypoints (Unconditioned) produces only a marginally worse flattened representation than our full approach, suggesting that the training objective already enforces the hierarchy even without this architectural bias. For reference, we train our keypoint encoder to minimize MSE loss with respect to ground-truth keypoints to show the upper-bound performance of our method. With the same number of keypoints (17) as ground-truth, this yields a RMSE error of 37.97.

On our new B&O dataset, which contains deformable objects, exhaustively annotating with all keypoints required to recover the full object configuration is intractable, since deformable objects have infinite degrees of freedom. We coarsely annotate a small test data subset and validate that Keypoint Pyramids performs better than baselines. We report these results in Sec A.2.

Utility for Downstream Tasks: H3.6M Action Category Recognition. Having established that Keypoint Pyramids discovers better keypoint representations than prior approaches, we now ask: how much do these representations contribute to downstream tasks? Towards evaluating this, we design an action

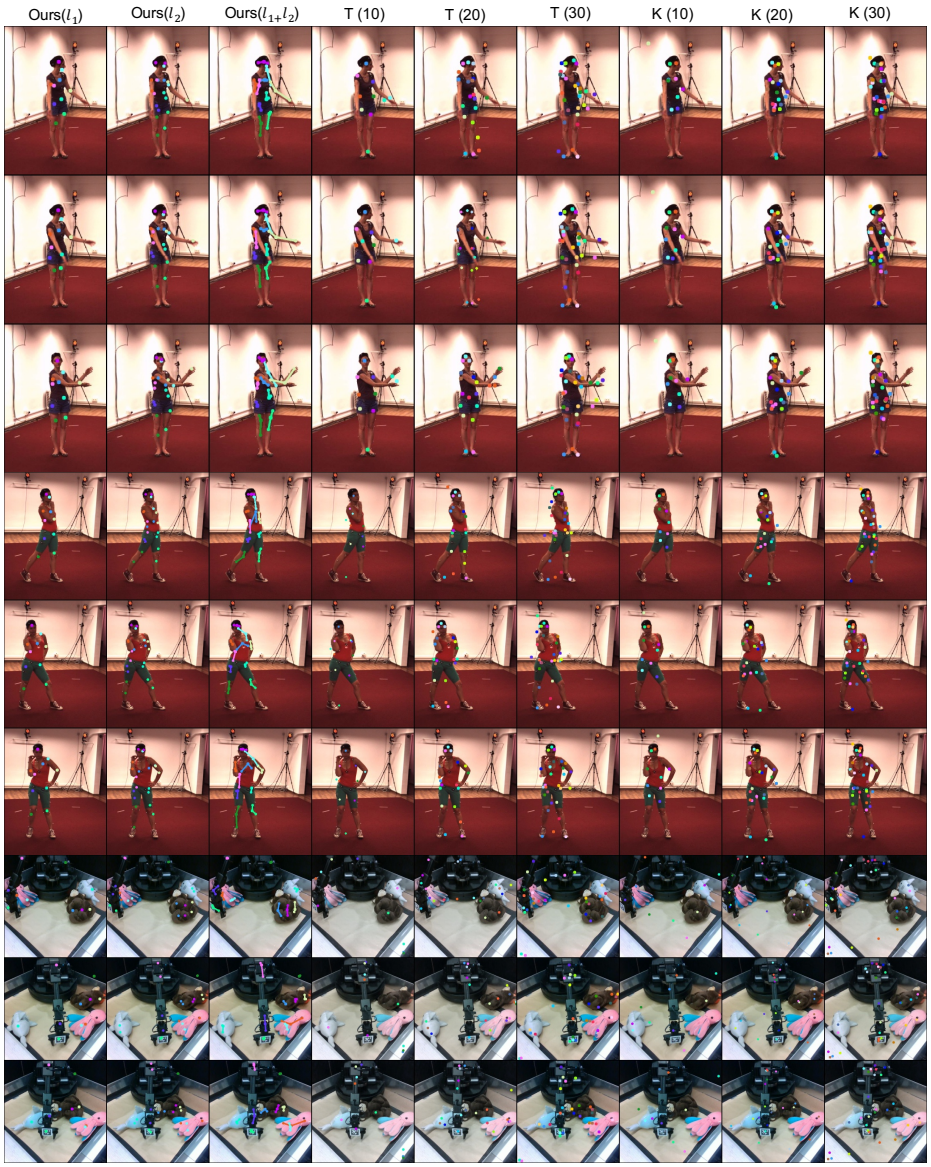

Fig. 6. (Best seen in pdf) Visualizing discovered keypoints from our method and baselines Transporter [24] (T) and KeyNet [19] (K). Each row is a single image and each column is a method. For our method, parent and children keypoints are illustrated with the same color, and their connections are drawn in column 3. More results in appendices.

classification task for recognizing activities from sequences of human poses. For H3.6M, where videos come with 11 action category labels, we evaluate discovered keypoint representations as inputs for training GRU-based recurrent networks for action classification. We describe the detail of these networks in Sec A.5.

As shown in Table 2, Keypoint Pyramids performs substantially better than the two baselines for this task, either by using the full flattened keypoint representations(L1+L2) or by just using the individual levels of keypoints. The first row of Table 2 shows an upper bound for this task: the action classification accuracy when input features are the 17 ground-truth keypoints. These results show that the improved quality of our discovered keypoint representations confers benefits for downstream tasks that use those representations.

Keypoint Visualizations. Figure 6 visualizes discovered keypoints on both datasets for our method (levels 1, 2, and combined) and the baselines Transporter (T) and KeyNet (K) trained with varying keypoint counts. We observe that Keypoint Pyramids recovers meaningful keypoints and hierarchies. On H3.6M, it discovers one coarse keypoint on each knee (yellow, cyan), connected pyramidally to two fine keypoints above and below capturing the full leg pose, a similar elbow pyramid (green) to capture the configuration of an arm, and a pyramid centered at the hip (light blue) that captures the relative orientation of the torso to the lower body. Even when discovered keypoints do not map one-to-one to semantic keypoints, they are consistently located on the body, and bind to specific locations, for example, the green pyramid near the right shoulder, and the pink pyramid near the top of the head. On the other hand, the flat baselines bind less consistently: for example, Transporter scatters many keypoints around the body rather than on it (column 6), and KeyNet produces keypoints that switch positions between actors or poses. In addition to these visualizations, we also train separate decoders to map discovered keypoint coordinates to image reconstructions. On both datasets, we see a clear progression of image reconstruction quality from level 1 to 2 to combination. Details in Sec A.4.

5 Conclusions

We have presented Keypoint Pyramids, an approach to tackle the challenging task of discovering coarse-to-fine keypoint hierarchies from unlabeled images. Keypoint Pyramids is designed to meet three key desiderata of comprehensiveness, graded informativeness, and parent-child associations between levels. Our results show the first examples of successfully discovered keypoint hierarchies, and our flattened representations outperform prior state-of-the-art for keypoint discovery.

Acknowledgements: This work was partially supported by an Amazon Research Award to Dinesh Jayaraman.

Bibliography

[1] Christopher P. Burgess, Loïc Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matthew M. Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation. *ArXiv*, abs/1901.11390, 2019.

[2] Boyuan Chen, Pieter Abbeel, and Deepak Pathak. Unsupervised learning of visual 3D keypoints for control. June 2021.

[3] Qi Dang, Jianqin Yin, Bin Wang, and Wenqing Zheng. Deep learning based 2d human pose estimation: A survey. *Tsinghua Science and Technology*, 24 (6):663–676, 2019.

[4] Neha Das, Sarah Bechtle, Todor Davchev, Dinesh Jayaraman, Akshara Rai, and Franziska Meier. Model-based inverse reinforcement learning from visual demonstrations. *CORL*, 2020.

[5] Martin Engelcke, Oiwi Parker Jones, and Ingmar Posner. Reconstruction bottlenecks in Object-Centric generative models. July 2020.

[6] Martin Engelcke, Adam R. Kosiorek, Oiwi Parker Jones, and Ingmar Posner. Genesis: Generative scene inference and sampling with object-centric latent representations. *ArXiv*, abs/1907.13052, 2020.

[7] Pedro Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. Ieee, 2008.

[8] Pedro F Felzenszwalb and Daniel P Huttenlocher. Pictorial structures for object recognition. *Int. J. Comput. Vis.*, 61(1):55–79, 2005.

[9] Pedro F Felzenszwalb and Daniel P Huttenlocher. Pictorial structures for object recognition. *International journal of computer vision*, 61(1):55–79, 2005.

[10] Martin A Fischler and Robert A Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on computers*, 100(1):67–92, 1973.

[11] Golnaz Ghiasi and Charless C Fowlkes. Occlusion coherence: Localizing occluded faces with a hierarchical deformable part model. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2385–2392, 2014.

[12] Anand Gopalakrishnan, Sjoerd van Steenkiste, and Jürgen Schmidhuber. Unsupervised object keypoint learning using local spatial predictability. *ArXiv*, abs/2011.12930, 2021.

[13] Anand Gopalakrishnan, Sjoerd van Steenkiste, and Jürgen Schmidhuber. Unsupervised object keypoint learning using local spatial predictability. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=GJwMHetHc73`.

[14] Klaus Greff, Raphael Lopez Kaufman, Rishabh Kabra, Nicholas Watters, Christopher P. Burgess, Daniel Zoran, Loïc Matthey, Matthew M.

Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. *ArXiv*, abs/1903.00450, 2019.

[15] Klaus Greff, Sjoerd Van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

[16] Shaoli Huang, Mingming Gong, and Dacheng Tao. A coarse-fine network for keypoint localization. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3047–3056, 2017.

[17] Shaoli Huang, Mingming Gong, and Dacheng Tao. A coarse-fine network for keypoint localization. In *Proceedings of the IEEE international conference on computer vision*, pages 3028–3037, 2017.

[18] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, 2014. https://doi.org/10.1109/TPAMI.2013.248.

[19] Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks through conditional image generation. In *NeurIPS*, 2018.

[20] Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Self-supervised learning of interpretable keypoints from unlabelled videos. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8784–8794, 2020.

[21] Jindong Jiang, Sepehr Janghorbani, Gerard de Melo, and Sungjin Ahn. Scalable object-oriented sequential generative models. *CoRR*, abs/1910.02384, 2019. URL `http://arxiv.org/abs/1910.02384`.

[22] Sheng Jin, Wentao Liu, Enze Xie, Wenhai Wang, Chen Qian, Wanli Ouyang, and Ping Luo. Differentiable hierarchical graph grouping for multi-person pose estimation. In *European Conference on Computer Vision*, pages 718–734. Springer, 2020.

[23] Thomas Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models. *ArXiv*, abs/1911.12247, 2020.

[24] Tejas D Kulkarni, Ankush Gupta, Catalin Ionescu, Sebastian Borgeaud, Malcolm Reynolds, Andrew Zisserman, and Volodymyr Mnih. Unsupervised learning of object keypoints for perception and control. *Advances in neural information processing systems*, 32, 2019.

[25] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017.

[26] Mike Lambeta, Po-Wei Chou, Stephen Tian, Brian Yang, Benjamin Maloon, Victoria Rose Most, Dave Stroud, Raymond Santos, Ahmad Byagowi, Gregg Kammerer, Dinesh Jayaraman, and Roberto Calandra. Digit: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation. *ICRA and IEEE RA-L*, 2020.

[27] Sijin Li and Antoni B. Chan. 3d human pose estimation from monocular images with deep convolutional neural network. In *ACCV*, 2014.

[28] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[29] 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. *CoRR*, abs/2001.02407, 2020. URL `http://arxiv.org/abs/2001.02407`.

[30] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *ArXiv*, abs/2006.15055, 2020.

[31] Dominik Lorenz, Leonard Bereska, Timo Milbich, and Björn Ommer. Unsupervised part-based disentangling of object shape and appearance. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10947–10956, 2019.

[32] Lucas Manuelli, Yunzhu Li, Peter R. Florence, and Russ Tedrake. Keypoints into the future: Self-supervised correspondence in model-based reinforcement learning. In *CoRL*, 2020.

[33] Alexander Mathis, Pranav Mamidanna, Kevin M Cury, Taiga Abe, Venkatesh N Murthy, Mackenzie Weygandt Mathis, and Matthias Bethge. Deeplabcut: markerless pose estimation of user-defined body parts with deep learning. *Nature neuroscience*, 21(9):1281–1289, 2018.

[34] Ajmal S Mian, Mohammed Bennamoun, and Robyn Owens. Keypoint detection and local feature matching for textured 3d face recognition. *International Journal of Computer Vision*, 79(1):1–12, 2008.

[35] Matthias Minderer, Chen Sun, Ruben Villegas, Forrester Cole, Kevin Murphy, and Honglak Lee. Unsupervised learning of object structure and dynamics from videos. June 2019.

[36] Matthias Minderer, Chen Sun, Ruben Villegas, Forrester Cole, Kevin P. Murphy, and Honglak Lee. Unsupervised learning of object structure and dynamics from videos. *ArXiv*, abs/1906.07889, 2019.

[37] Damian Mrowca, Chengxu Zhuang, Elias Wang, Nick Haber, Li F Fei-Fei, Josh Tenenbaum, and Daniel L Yamins. Flexible neural representation for physics prediction. *Advances in neural information processing systems*, 31, 2018.

[38] Despoina Paschalidou, Angelos Katharopoulos, Andreas Geiger, and Sanja Fidler. Neural parts: Learning expressive 3d shape abstractions with invertible neural networks. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3203–3214, 2021.

[39] Georgios Pavlakos, Xiaowei Zhou, Aaron Chan, Konstantinos G Derpanis, and Kostas Daniilidis. 6-dof object pose from semantic keypoints. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 2011–2018. IEEE, 2017.

[40] Zengyi Qin, Kuan Fang, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. KETO: Learning keypoint representations for tool manipulation. October 2019.
[41] Nermin Samet and Emre Akbas. Hprnet: Hierarchical point regression for whole-body human pose estimation. *ArXiv*, abs/2106.04269, 2021.
[42] Aliaksandr Siarohin, Subhankar Roy, Stéphane Lathuilière, S. Tulyakov, Elisa Ricci, and N. Sebe. Motion-supervised co-part segmentation. *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9650–9657, 2021.
[43] Aleksandar Stanić, Sjoerd van Steenkiste, and Jürgen Schmidhuber. Hierarchical relational inference. *arXiv preprint arXiv:2010.03635*, 2020.
[44] James Thewlis, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks by factorized spatial embeddings. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3229–3238, 2017.
[45] Rishi Veerapaneni, John D. Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua B. Tenenbaum, and Sergey Levine. Entity abstraction in visual model-based reinforcement learning. *ArXiv*, abs/1910.12827, 2019.
[46] Haoyu Xiong, Quanzhou Li, Yun-Chun Chen, Homanga Bharadhwaj, Samarth Sinha, and Animesh Garg. Learning by watching: Physical imitation of manipulation skills from human videos. *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7827–7834, 2021.
[47] Zhenjia Xu, Zhijian Liu, Chen Sun, Kevin P. Murphy, William T. Freeman, Joshua B. Tenenbaum, and Jiajun Wu. Unsupervised discovery of parts, structure, and dynamics. *ArXiv*, abs/1903.05136, 2019.
[48] Jingyun Yang, Junwu Zhang, Connor Settle, Akshara Rai, Rika Antonova, and Jeannette Bohg. Learning periodic tasks from human demonstrations. *ArXiv*, abs/2109.14078, 2022.
[49] Y. Zhang, Yijie Guo, Yixin Jin, Yijun Luo, Zhiyuan He, and Honglak Lee. Unsupervised discovery of object landmarks as structural representations. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2694–2703, 2018.
[50] Daniel Zoran, Rishabh Kabra, Alexander Lerchner, and Danilo Jimenez Rezende. Parts: Unsupervised segmentation with slots, attention and independence maximization. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10419–10427, 2021.