Conditional Link Prediction of Category-Implicit Keypoint Detection

Ellen Yi-Ge[1], Rui Fan[2], Zechun Liu[1], Zhiqiang Shen[1]
[1] Carnegie Mellon University
[2] UC San Diego

Abstract

Keypoints of objects reflect their concise abstractions, while the corresponding connection links (CL) build the skeleton by detecting the intrinsic relations between keypoints. Existing approaches are typically computationally-intensive, inapplicable for instances belonging to multiple classes, and/or infeasible to simultaneously encode connection information. To address the aforementioned issues, we propose an end-to-end category-implicit Keypoint and Link Prediction Network (KLPNet), which is the first approach for simultaneous semantic keypoint detection (for multi-class instances) and CL rejuvenation. In our KLPNet, a novel Conditional Link Prediction Graph is proposed for link prediction among keypoints that are contingent on a predefined category. Furthermore, a Cross-stage Keypoint Localization Module (CKLM) is introduced to explore feature aggregation for coarse-to-fine keypoint localization. Comprehensive experiments conducted on three publicly available benchmarks demonstrate that our KLPNet consistently outperforms all other state-of-the-art approaches. Furthermore, the experimental results of CL prediction also show the effectiveness of our KLPNet with respect to occlusion problems.

1. Introduction

Accurate semantic keypoint localization and detection is the basic prerequisite for copious computer vision applications, including simultaneous localization and mapping [20], human pose estimation [9], hand key-joint estimation [16], *etc*. The connection link provides an additional semantic relation between each pair of keypoints and it can be used for many semantic-level tasks. Nevertheless, prevailing keypoint detection methods, *e.g.*[13, 21], mainly focus on multi-person human pose estimation, which aims at recognizing and localizing anatomical keypoints (or body joints) and human skeleton connections. On the other hand, existing object (or rigid body) keypoint localization approaches, *e.g.*[19, 29, 22, 18], always fail to successively explore CL among keypoints. Few studies tackled the prob-
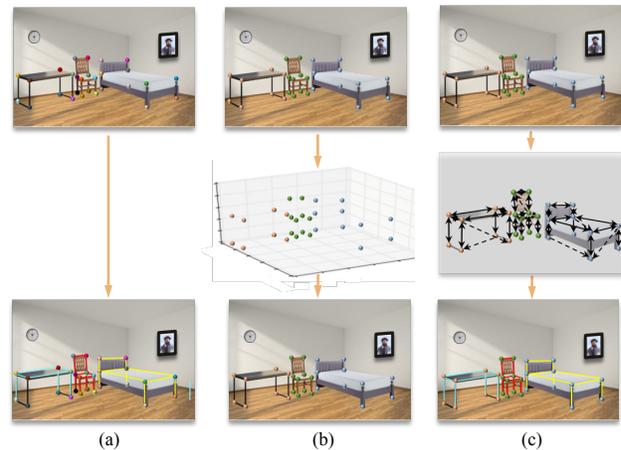
Figure 1: (a) KeypointNet [22]. (b) StarMap [30]. (c) Our proposed Keypoint and Link Prediction Network (KLPNet), a category-implicit approach. KLPNet is the first framework capable of finding connection link for multi-class object keypoints. KLPNet adopts a conditional embedding graph to implement the link prediction based on the extracted features from the 2D images.

lem of simultaneously inferring keypoint and encoding their semantic connection information for multi-class instances.

Inspired by several prevalent bottom-up approaches in the field of multi-person pose estimation, such as [2, 7, 17], which directly localize all keypoints from multi-instances and group keypoints into persons to find skeleton connection, we wonder what if to apply a similar mechanism on multi-class object keypoint detection and CL prediction. However, the previous methods cannot be simply grafted onto this issue. The conventional approaches stack each heatmap as a particular class of keypoint for single-type pose estimation, especially for classes of nodes in human pose estimation which belongs to only one category: person. When we move to multi-class rigid bodies, the conventional approaches are inefficient and costly, even impossible, since each category contains numerous classes of keypoints. Consequently, two key factors should be addressed: *(1) how to deal with multi-class instances* and *(2) how to encode the semantic keypoints and their connection links*. Since geometric contextual relations are the keys to iden-

tify the keypoint-instance affiliation, these relations spontaneously construct a graph, which consists of nodes (keypoints) and edges (relations between keypoints). A graph structure could be constructed. Besides, the crucial semantic information – the category of instance that a given keypoint belongs to, should also be considered for CL prediction for multi-class objects. So we can then discover geometrically and semantically consistent keypoints across instances of different categories.

There are two mainstream approaches in instance keypoint detection field, category-specific detection and category-agnostic detection. For category-specific methods like KeypointNet [22] in Fig. 1(a), they typically group each keypoint as an independent category concerning a given classified target, which is extremely ineffective, costly, and practically inapplicable for building a graph for keypoint and CL prediction for objects with a varying number of parts. For category-agnostic methods like SVK [4] and StarMap [30], StarMap in Fig. 1(b) typically introduces a hybrid representation of all keypoints and compress those belonging to one target as the same class to a single heatmap, and then lift them into 3D space to adjust their locations in 2D. However, StarMap is also costly due to the massive 3D geometry information, such as depth map or multi-view consistency. Besides, the connection information is lost due to the lack of semantic property. Thus, we hope to find a novel, economical, yet powerful approach to directly work in the 2D images.

To this end, we propose a category-implicit method, Keypoint and Link Prediction Network (KLPNet), as shown in Fig. 2, including a Deep Path Aggregation Detector (DPAD), a Cross-stage Keypoint Localization Module (CKLM) and a Conditional Link Prediction Graph Module (CLPGM). For the first time, we implement the semantic keypoint detection without converting 2D information to 3D spaces, by virtue of the conditional graph neural network, on rigid bodies. CLPGM recovers the links straightforwardly based on the single heatmap and implicit features of each target extracted from the module DPAD. In CKLM, the Cross-stage feature aggregation scheme is proposed to overcome the ambiguous locations of the category-implicit keypoints on the single heatmap. Specifically, a Location Instability Strategy (LIS) is utilized in GLPGM to disentangle occlusion cases and further respond the defective keypoint localization to the previous module, CKLM.

The main contributions of this paper can be summarized as follows: (1) To the best of our knowledge, we present the first category-implicit and graph embedding approach, KLPNet, to effectively construe keypoints for instances under multiple categories with a flexible number of semantic labels, and further predict its conditional connection links (2) We propose the keypoint data representation without redundant 3D geometry information, whose location can

be adjusted through Cross-stage feature aggregation in a coarse-to-fine manner (3) A novel link prediction module, CLGPM, can enhance the node links based on the single heatmap and extracted implicit features of each target, providing the geometric and semantic information for link recovery. An innovative strategy, LIS in CLGPM, are capable of disentangling the cases with occlusions (4) We explore a deep path aggregation detector to localize the targeting instances precisely.

2. Related Work

2.1. Keypoint Estimation and Geometric Reasoning

The prominent thoughts of keypoint detection on rigid bodies concentrate on the feature extraction as a two-stage pipeline: identify the localization of each object on the image, and then solve the single object pose estimation problem based on the cropped target. Stacked hourglass [15] stacks hourglasses that are down-sampled and up-sampled modules with residual connections to enhance the pose estimation performance. Based on the stacked hourglass network, Cascade Pyramid Network[3] address the pose estimation by adopting two sub-networks: GlobalNet and RefineNet. GlobalNet locates each keypoint to one heatmap that is easier to detect, then RefineNet explicitly address the 'hard' keypoints that requires more context information and processing rather than the nearby appearance feature. The multi-stage pose estimation network (MSPN) [9] extends the GlobalNet to multiple stages for aggregating features across different stages to strengthen the information flow and mitigate the difficulty in training.

The conventional approaches stack each heatmap as a special class of keypoint for single types, including as left-top, right-top, left-bottom, right-bottom ones. Based on well-defined semantic keypoints, when we move to keypoint detection on multi-class objects, such as bus, chair, ship etc, it is ineffective and costly to train $N \times C$ classes keypoints, where N represents the total number of keypoints of each category, and C is the number of categories. In addition, the value of N varies in different categories. In terms of merging keypoints from multiple targets, consistent correspondences should be established between different keypoints across multiple target categories, which is difficult or sometimes impossible. Besides, category-specific keypoint encoding fail to capture both the intra-category part variations and the inter-category part similarities.

To solve the above-mentioned issues, category-agnostic approaches project the keypoints that belong to the same target to the same category on one heatmap, and then provide additional information to convert 2D image to 3D space for pose estimation. StarMap [30] mixes all types of keypoints using a single heatmap for general keypoint detection. KeyPointNet [22] considers the relative pose es-
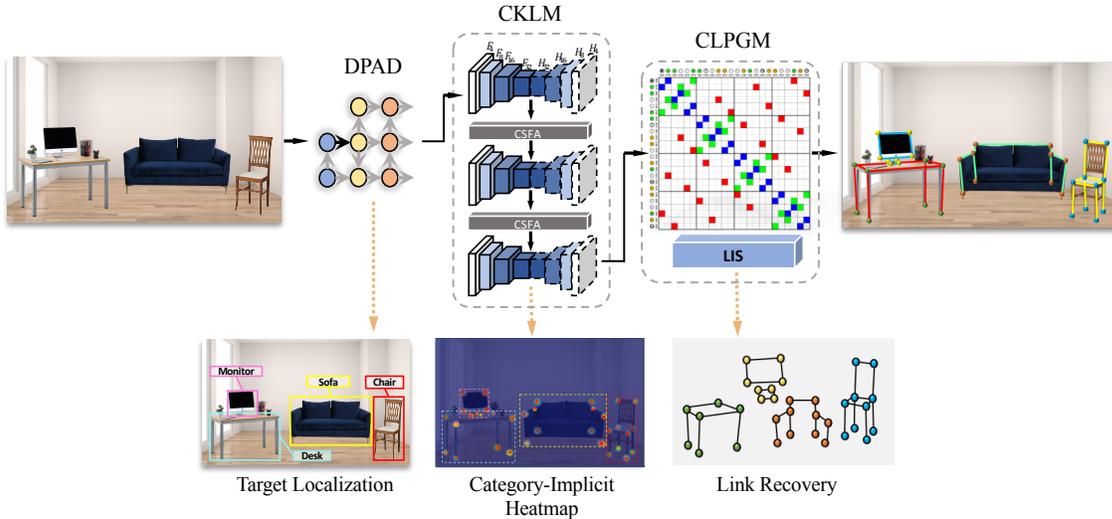
Figure 2: Illustration of the proposed KLPNet that is composed of DPAD, CKLM (detailed in Fig. 3) and CLPGM. After objects are allocated by DPAD, category-implicit nodes that belong to the same target are classified as one class on the single heatmap by Cross-stage feature aggregation (CSFA, detailed in Fig. 4). F and H are down-sampling and up-sampling feature maps with subscript representing their size. CLPGM works on the nodes with extracted features and corresponding labels to rejuvenate the node links. The Location Instability Stragety (LIS, detailed in Fig. 5) of the inference nodes helps tackle occlusion issues and provide CKLM with feedback to rectify keypoint localization.

timation loss to penalize the angular difference between the ground truth rotation and the predicted rotation using orthogonal procrustes. Both approaches have to convert the 2D to 3D space first, and then adjust the keypoint location by different predefined 3D models. Our approach generate one type of such general implicit keypoints with more explicit geometry property and top-class label. Besides, the geometric adjustment works in multiple stages in 2D space, increasing the cost-efficiency. Therefore, we consider a novel approach to skip 3D estimation and localize the keypoints more accurately.

2.2. Graph Link Prediction

There is a growing interest in the Graph Neural Network (GNN) because of its flexible utilization with body joint relations. [6] introduced the variational graph autoencoder (VGAE) for unsupervised learning on graph-structured data. [8] proposed a model called the Relational-variational Graph AutoEncoder (RVGAE) to predict concept relations within a graph consisting of concept and resource nodes. We propose a new graph structure, CLPGM, to predict the connection among keypoints with different object labels.

3. Methodology

3.1. Deep Path Aggregation Detector

Inspired by PANet [12], we propose DPAD, a Deep Path Aggregation Detector to enhance the localization capability of the entire feature hierarchy by propagating strong re-

sponses of low-level patterns. We refer the readers to **supplementary material** for its architecture. ResNeXt [27] is used as the backbone to generate different levels of feature maps $C3 \sim C7$. In addition to these generated feature maps from FPN [10], $C8$ and $C9$, two higher-level feature maps, are created by down-sampling from $C7$. The augmented path starts from the lowest level and gradually approaches to the top. From $C3$ to $C9$, the feature map is down-sampled with factor 2. $\{N3 \sim N9\}$ denote newly generated feature maps corresponding to $\{C3 \sim C9\}$. Each building block takes a higher-resolution feature map N_i and a coarser map C_{i+1} through lateral connection and generates the new feature map N_{i+1}. Furthermore, we adopt CIoU [28] to penalize the union area over the circumscribed rectangle's area in IoU Loss. CIoU can improve the trade-off between speed and accuracy for bounding box (BBox) regression problems, and suppresses redundant BBox to increase the robustness of detector for occlusions.

3.2. Cross-stage Keypoint Localization

After object targeting, a CKLM would generate detailed localization of all category-implicit keypoints for each classified candidate.

3.2.1 Single-stage Mechanism

The backbone of the single-stage mechanism is ResNext-101 [27]. The shallow features have a high spatial resolution for localization but low semantic information for recognition. On the other hand, deep feature layers have rich se-
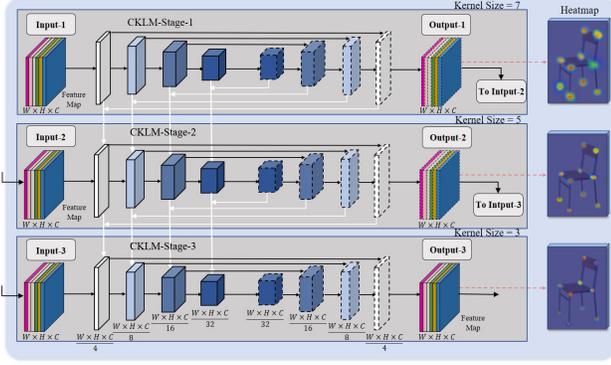
Figure 3: An overview of CKLM that consists of three stages with a coarse-to-fine surveillance strategy. A cross-stage aggregation scheme is adopted between adjacent stages (detailed in Fig. 4). The coarse-to-fine surveillance strategy utilizes distinctive Gaussian kernel size to boost the keypoint localization performance, as demonstrated on the heatmaps. W, H, C denote the height, weight and channel size for features respectively.

mantic information but lower spatial resolution due to the convolution and pooling layers. As shown in Fig. 3, both spatial resolution and semantic features from distinctive layers are integrated to avoid the unconscious information.

Since we hope to allocate all keypoints from the same target to a single heatmap, they are compelled to set as one class. This single-channel heatmap encodes the image locations of the underlying points. It is motivated by using one heatmap to encode occurrences of one keypoint on multiple persons. The keypoint heatmaps is combined with the confidence maps H and offset maps $\{O_x, O_y\}$. We adopt the binary cross-entropy loss to learn confidence maps with each targeting category and the Smooth L1 loss to update the offset maps.

$$\mathcal{L}_{kd} = \sum_i^c \sum_j^k (\Theta \delta(H - H^*) + \Upsilon \rho(O_{xy} - O_{xy}^*)), \quad (1)$$

where $\mathcal{L}_{kd}$ is the keypoint detection loss, δ is the binary cross-entropy loss, ρ is the Smooth L1 loss, Θ and Υ indicate the corresponding weights, c is the targeting categories, k denotes the number of keypoints under each targeting category. H^* and O_{xy}^* are the ground truth.

3.2.2 Cross-stage Feature Aggregation among Multiple Stages

Feature aggregation could be regarded as an extended residual design in the single-stage mechanism, which is helpful for dealing with the gradient vanishing problem. After the first single-stage module, the single heatmap contains most probable keypoint locations. However, the heatmap from the first single stage is a coarse prediction with abounding noise, even if adequate features have been already extracted
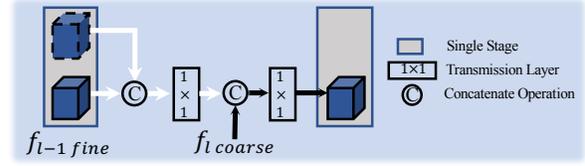
in the stage. Even small localization errors would significantly affect the keypoint detection performance. To filter the noise, another two stages are cascaded with the refined surveillance, as shown in Fig. 3. Since the Gaussian kernel is used to generate the ground truth heat map for each key point, we decide to utilize distinguishable sizes of kernels, 7, 5, and 3, in these three stages. This strategy is based on the observation that the estimated heat maps from multi-stages are also in a similar coarse-to-fine manner.

Fig. 4 shows the cross-stage feature aggregation scheme among multiple stages. f_l represents the extracted features in layer l. Since we hope to aggregate features from neighbour stages and layers, the coarse-to-fine approach is proposed as follows:

$$f_{l_{fine}} = T(C(f_{l_{coarse}}, f_{l-1_{out}})), \quad (2)$$

$$f_{l-1_{out}} = T(C(f_{l_{fine\_down}}, f_{l-1_{fine\_up}})), \quad (3)$$

where T is the transmission layer which is a 1×1 convolution operation, and C is the concatenation operation. The implementation details of CKLM are reported in **supplementary material**.

3.3. Conditional Link Prediction Graph Module

Since all the predefined keypoints are localized with agnostic category on the heatmap, we cannot directly connect them. To solve this issue, a CLPGM is considered to figure out the connection among each key point. In other words, given each target label, we hope to find the adjacency matrix when inferring on an unsupervised learnable graph. Our proposed CLPGM is a pioneer to explore connection links of category-implicit keypoints for multi-class objects.

3.3.1 Notations

We consider the keypoints under each targeting categories as the node of an undirected, unweighted graph $G = (V, E)$ with $N = |V|$ nodes. Each keypoint is seen as an individual node on gragh G. The features of each keypoint extracted from the previous stage are set as the node features and summarized in an $N \times D$ matrix X_C with the target label C, where D is the degree matrix of the graph. The diagonal elements of the graph's adjacency matrix A are set to

Figure 4: Cross-stage feature aggregation scheme. After concatenation, a transmission layer is applied to the features obtained from the previous stage before feature aggregation.

1, which means every node is connected to itself. The adjacency matrix A consists of several A_C. The stochastic latent variables z_i is summarized in an $N \times F$ matrix Z, where N is the number of keypoints, F is the depth of features, Z represents the embedding space.

3.3.2 Objective Function

CLPGM is built in a top-down manner. Given the training graph, we first disentangle all nodes with C types of labels, where C is the number of target categories. When the training samples from the CKLM arrive, the corresponding nodes with the same target label C are activated and their features X_C are updated. After retaining the rest nodes with their features, the whole graph updates the weights and learns the connection. The objective of learning the graph is given as follows:

$$argmax_{\theta_G} \prod P(A_C|X_C, \theta_G). \quad (4)$$

CLPG learns node parameters θ_G that best fits feature maps of training images.

Let us regard X_C as a distribution of "node activation entities". We consider the node response of each unit $x \in X_C$ as the number of "node activation entities". $F(X) = \tau \cdot max(f_x, 0)$ denotes the number of updated nodes, where $f(X)$ is the normalized response value of x and τ is a constant parameter.

For a Gaussian mixture model, the distribution of the whole graph at each step is as follows:

$$P(A_C|X_C, \theta_G) = \prod_{X_C \in X} q(p_{A_C}|X_C, \theta_G)^{F(x)}, \quad (5)$$

where $q(p_{A_C}|X_C, \theta_G)$ indicates the compatibility of each part of updated sub-adjacency matrix.

Each time B targets are detected from the sample training image, the corresponding sub-adjacency matrices are updated to the whole graph for the advanced training. The features are projected to an embedding space, Z, after the first four layers of CLPG. The feature information could be encoded as:

$$q(Z|X, A, C) = \prod_{i=1}^{N} q(z_i|X, A, C), \quad (6)$$

where $q(z_i|X, A, C) = Gaussian(z_i|c_i, \mu_i, diag(\sigma_i^2))$. In the Equation 6, μ and σ^2 are the matrices of expected value and variance. Each layer is defined as $tanh(\hat{A}XW_i)$. Denote the target category prediction as c, we calculate the piecewise link loss function as follows:

$$\mathcal{L}_{link} = \begin{cases} (c - c_i)^2 & c \neq c_i \\ E_\varrho log_p[A|Z] - KL[\varrho\,||\,p(Z)] & c = c_i \end{cases}$$

where $\varrho = q(Z|X, A)$, $KL[q(\cdot)||p(\cdot)]$ represents the Kullback-Leibler divergence between $q(\cdot)$ and $p(\cdot)$.

3.3.3 Rejuvenation

For a non-probabilistic variant of the C-GVAE model, we calculate embeddings Z and the rejuvenated adjacency matrix $\hat{A}$ as follows:

$$\hat{A} = \sigma(ZZ^T|C), \quad (7)$$

where σ is the inner-product operation and C is the targeting category.

Thus, the final loss function of KLPNet can be formulated as:

$$\mathcal{L}_{KLPNet} = \alpha \mathcal{L}_{kd} + \beta \mathcal{L}_{link}, \quad (8)$$

where α and β are the predefined constant parameters.

3.3.4 Location Instability Strategy of Inference Nodes

When multiple targets are occluded, the number of detected nodes may be higher than the predefined number in an area. If these targets belong to distinctive categories, this issue is simplified to generate multi singular heatmaps for each target category. However, if these targets are with the same label, we should design a location instability strategy to infer which nodes are for each overlapped target. As illustrated in Fig. 5(a), an occlusion appears at the center of the image. After keypoint localization through the CKLM, the generated heatmap is illlustrated in the Fig. 5(b). The red and yellow keypoints named fixed nodes are identified to each target. However, the white keypoints with category-implicit nodes on the intersection region (shade part) are confused to be categorized into either of the two monitors to the particular target.

We first assume that the category-implicit keypoints of the target with the same label share similar features. For example, the similarities of the features belonging to two top-right keypoints of monitors have higher probability of coming from the same instance. Next, We considered that if certain special adjoining nodes always triggered a node, then the inferred node's distance and certain fixed nodes of the object part should not change greatly among targets with the same category label. In the inference part, partial implicit nodes and fixed nodes are utilized to complete the link prediction, where the number of the total nodes should equal to the predefined number of each target category. As shown in the Fig. 5(d), two cases of the predicted results are given, and the incorrect link prediction (red dash line) is deleted. Thus, the implicit nodes are distinguished as inferred node and outlier, as illustrated in Fig. 5(c).

4. Experiments and Results

4.1. Setting

Datasets Our models are evaluated on MSCOCO dataset [11], Pascal 3D+ [26] and Object-Net3D [25]. MSCOCO
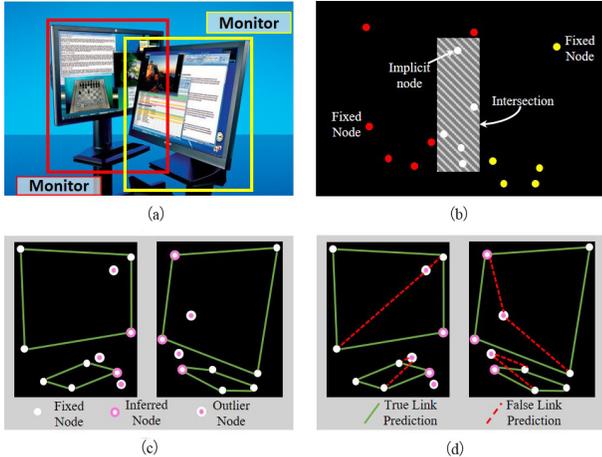
Figure 5: Location Instability Strategy. (a) two targets with the same label (monitor) appear in the image, and there exists an occlusion area; (b) the generated heatmap contains three types of nodes (or keypoints): red and yellow nodes are the irrefutable ones which belong to the two targets separately; white nodes are the debatable ones localized in the occlusion area; (c) due to location instability, the debatable ones are distinguished as outliers; (d) illustrations of true (green solid lines) and false (red dash lines) link predictions.

Table 1: Detection performance comparison of different models with backbones including ResNet-101 [5], ResNext-101 [27] and [24] on MSCOCO Dataset.

Model	AP	AP_{50}	AP_{75}	Backbone
Mask R-CNN +FPN [10]	35.7	58.0	37.8	ResNet-101
Mask R-CNN +FPN [10]	37.1	60.0	39.4	ResNeXt-101
PA-Net [12]	42.0	65.1	45.7	ResNeXt-101
Yolo v4 + PA-Net[1]	42.3	64.3	45.7	CSPResNeXt50
Ours (DPAD)	**42.7**	**64.7**	**45.8**	ResNeXt-101

Table 2: Results of the backbone with different layers on MSCOCO Dataset.

Method	ResNeXt-50	ResNeXt-101	ResNeXt-152
AP	40.6	42.7	43.3
FLOPs (G)	4.6	7.3	12.1

includes 118k images and 860k annotated objects. We split the train and test dataset as 18K and 2K images, which belongs to 20 types of rigid bodies, such as car, bus, chair, desk etc. Pascal3D+ contains 12 man-made object categories with 2K to 4K images per category. We make use of its category-implicit 2D keypoints for experiments. For fair comparison with state-of-arts, evaluation is done on the subset of the validation set that is non-truncated and non-occluded. Object-Net3D contains 50k samples, while 20k ones have keypoint annotations. We use 19k images with annotation files as the training set and 10k images for test datasets.

Evalutation Metrics The main metric is Average Precision (AP) over a dense set of fixed recall threshold. We show AP at Intersection-over-Union (IoU) or Object Keypoint Similarity (OKS) threshold 50% (AP_{50}), 75%(AP_{75}), and averaged over all thresholds between 0.5 and 1. For keypoint localization, two protocols, Percentage of Correct Keypoints (PCK) and Oracle assigned Keypoint Identification (OracleId), are considered to evaluate the performance of the models. Floating Point Operations per Second (FLOPs) is used to measure computational performance.

Implementation Details The KLPNet is implemented in the PyTorch framework and trained on four Nvidia RTX 2080 Ti GPU in 180k iterations. We adopted Mosaic to combines four training images with different scales to one image. In addition, batch normalization calculates activation statistics from four different images on each layer. This significantly reduces the need for a large mini-batch size.

4.2. Analyses

Choice of Backbone for Detector Since detection boxes are critical for top-down approaches in object estimation, we compare our model with the other four models who has similar backbone, as demonstrated in Tab. 1. Based on ResNeXt-101, the corresponding APs of Mask R-CNN + FPN, PANet, and our model are 37.1%, 42.0%, and 42.7%,. Even comparing with Yolov4 + PAN whose backbone is CSPResNeXt50, the performance of our model is slightly better, since our layer is deeper. Tab. 2 demonstrates the AP and FLOPs of our backbone with different layers. The performance gets quickly saturated with the growth of backbone capacity. ResNeXt-101 outperforms ResNeXt-50 by 2.1% AP and consume 2.7G FLOPs more, but there is only 0.6% gain from ResNeXt-101 to ResNeXt-152 at the cost of additional 4.8G FLOPs. It is not effective to adopt ResNeXt-152 or larger backbones for a single-stage network.

Heatmap Manufacturing with CKLM Tab. 3 shows the performance comparison of the keypoint localization and classification among three state-of-the-art approaches. In the top of 3, $PCK, \alpha = 0.1$ analyze a detected joint is acceptable if the distance between the predicted and the ground-truth joint is within the threshold 0.1. The mean PCK of the evaluation of 6-DoF Stacked Hourglass (SH) and StarMap are 82.5% and 78.6%, and our model achieves 86.6%. Although the performance of our model is better, the increased values of PCK are only 4.1% and 8.0%. We emphasize that all counterpart methods are category-specific, thus requiring ground truth object category as input while ours is general. Thus, we did another experiment to prove our thought and identify the improvement of models. The bottom of the table 3 factors out the mismanagement caused by improper keypoint ID association. It is obvious that the score is improved to 96.4%. This is quite encouraging since

Table 3: 2D keypoint localization results on Pascal3D+. The results are shown in PCK ($\alpha = 0.1$). Top: our results with the nearest canonical feature as keypoint identification. Bottom: results with oracle keypoint identification. StarMap*: our re-implementation of StarMap [30] with ResNeXt-101 as the backbone for a fair comparison to ours. Absolute and relative improvements of mean AP (PCK: MCKP; OracleId: 6-DoF SH) are shown in column $\triangle$. CS, CA, CI represents category-specific, category-agnostic, category-implicit methods respectively.

PCK ($\alpha = 0.1$)	TYPE	aero	bike	boat	bottle	bus	car	chair	table	moto	sofa	train	tv	mean	$\triangle$
MCKP [23]	CS	66.0	77.8	52.1	83.8	88.7	82.3	65.0	47.3	68.3	58.8	72.0	65.1	68.8	-
6-DoF SH [19]	CS	84.1	86.9	62.3	87.4	96.0	93.4	76.0	-	-	78.0	58.4	84.8	82.5	+13.7
StarMap [30]	CA	75.2	83.2	54.8	87.0	94.4	90.0	75.4	58.0	68.8	79.8	54.0	85.8	75.5	+6.6
StarMap* [30]	CA	75.4	83.8	54.2	87.8	94.1	90.5	75.9	59.1	69.2	79.7	55.8	86.8	76.0	+7.2
Ours (CKLM)	CI	85.3	89.1	72.3	93.4	98.5	96.7	82.3	78.4	86.9	89.3	76.1	91.2	86.6	+17.8
OracleId	TYPE	aero	bike	boat	bottle	bus	car	chair	table	moto	sofa	train	tv	mean	
6-DoF SH [19]	CA	92.3	93.0	79.6	89.3	97.8	96.7	83.9	-	-	85.1	73.3	88.5	89.0	-
StarMap [30]	CA	93.1	92.6	84.1	92.4	98.4	96.0	91.7	90.0	90.1	89.7	83.0	95.2	92.2	+3.2
StarMap* [30]	CA	94.2	92.9	84.8	93.1	98.8	96.5	90.4	90.6	92.1	89.8	83.3	95.7	92.6	+3.7
Ours (CKLM)	CI	96.2	96.7	89.6	94.8	99.4	98.2	94.3	95.8	96.6	94.2	93.8	98.4	96.4	+7.5

Table 4: i -s represents i stages. Top: Performance of different models with single stage (including two-stage Hourglass). Bottom: our CKLM with different number of stages. All experiments are conducted on MSCOCO minival dataset.

	AP	FLOPs (G)
1-s Hourglass [14]	54.5	3.92
2-s Hourglass [14]	66.5	6.14
CPN-GlobalNet [3]	66.6	3.90
1-s MSPN [9]	71.5	4.4
Our CKLM (1-s)	72.0	5.1
Ours (2-s)	75.3	10.1
Ours (3-s)	76.1	15.3
Ours (4-s)	76.4	20.1
Ours (5-s)	76.9	23

our approach is designed to be a general purpose keypoint predictor on rigid body, indicating that it is advantageous to train a unified network to predict keypoint locations, as this allows to train a single network with more relevant training data.

Single Stage Mechanism for CKLM In this part, we demonstrate the effectiveness of CKLM based on the proposed single-stage mechanism. According to Tab. 4, the performance of our single-stage model with 72.0% AP on MSCOCO minival dataset demonstrates the superiority over others. Note that the localization of the keypoints are only considered since the points has no semantic information at this part.

Cross-stage with Feature Aggregation Scheme for CKLM From Tab. 4, the performance of single-stage Hourglass is inferior. Adding one more stage introduces a large AP margin. Thus, multi-stages of CKLM are further investigated. Tab. 4 shows the performance of divergent stages based on our single-stage model. By comparing with single stage, 2-stage CKLM further leads to a 2.3% improvement. Introducing the third, fourth and fifth stage maintains a tremendous upward trend and eventually brings an impressive performance boost of 0.8, 0.3 and 0.5 AP improvement on the previous stage. Meanwhile, the cost of three stages

is more than two stages 5.2 FLOPs, but an additional 4.8 FLOPs are consumed if the fourth stage is adopted. Finally, the three-stage CKLM is considered to balance the precision and the cost. These experiments indicate that our model successfully pushes the upper bound of existing single-stage and multi-stage CKLM. It obtains noticeable performance gain with more network capacity.

Performance of the Keypoint and Link Prediction Network We compared the performance among three approaches: StarMap, KeypointNet and our KLPNet on ObjectNet3D+. Tab. 5 illustrates the performance of the approaches mentioned above. Both StarMap and KeypointNet convert the image to the 3D spaces with different additional information, which is inefficient and costly. We did the experiment on KLPNet in three different views: the KLPNet with only backbone, KLPNet⋆ includes the cross-stage feature aggregation scheme; and KLPNet† contains both cross-stage feature aggregation scheme and location instability strategy. From Tab. 5, KLPNet† achieves the best performance on distinctive categories.

Visual Results of Keypoints and Connection Links Since our approach is the forerunner for link prediction on multi-class rigid bodies, it is hard to compare the quantitative and qualitative results with others. Here we visualize the conditional connection link to illustrate the qualitative performance. According to Fig. 6, our KLPNet† provides robust connection links in various cases. The semantic information well manifests themselves.

5. Conclusion

In this work, we proposed a novel and effective simultaneous multi-class object keypoint and connection link rejuvenation approach. The major contributions of this network include: 1) DPAD, a detector capable of localizing category-implicit keypoints accurately; 2) CLPGM, a novel link prediction module capable of recovering the node links straightforwardly based on a single heatmap and the implicit features of each target extracted by DPAP; 3) LIS,

Table 5: Comparison of keypoint estimation on ObjectNet3D+. Note: Ours represents KLPNet with only the basic backbone; Ours★ additionally includes the Cross-stage (three stages) feature aggregation scheme; Ours† contains both Cross-stage feature aggregation scheme and Location Instability Strategy (LIS). Definition of Type and StarMap* are informed in Tab. 3. Benchmark of column △ is StarMap.

	TYPE	bed	sofa	bookshelf	chair	monitor	cabinet	microwave	console	guitar	mean	△
StarMap [30]	CA	72.0	66.3	77.8	73.4	88.6	84.1	94.3	43.1	73.3	-	-
StarMap* [30]	CA	72.3	66.6	77.4	73.8	89.1	84.7	94.5	43.4	74.2	-	-
KeypointNet [22]	CS	74.9	71.2	74.3	76.8	81.4	88.7	95.4	57.6	71.3		
Ours	CI	69.6	68.9	71.8	74.1	79.6	84.3	91.4	63.7	71.6	-	-
Ours★	CI	81.3	75.8	76.3	77.6	85.3	87.9	96.1	68.5	74.8	-	-
Ours†	CI	**87.4**	**81.1**	**83.4**	**84.8**	**89.7**	**90.1**	**77.8**	**71.3**	**79.9**	-	-
	TYPE	car	bus	aircraft	mirror	piano	helmet	loudspeaker	knife	printer	mean	△
StarMap [30]	CA	58.7	74.8	63.5	67.9	57.1	69.7	56.7	18.2	63.4	67.0	-
StarMap* [30]	CA	58.9	75.1	63.6	67.4	57.6	70.2	56.9	18.4	63.6	68.0	1.0
KeypointNet [22]	CS	69.8	81.3	71.4	69.6	64.8	72.6	69.4	58.7	72.7	72.4	5.4
Ours	CI	63.1	76.9	69.0	69.1	63.5	72.1	68.8	64.1	71.3	71.9	4.9
Ours★	CI	69.7	80.1	73.3	73.5	69.8	75.3	73.9	69.3	74.8	76.8	9.8
Ours†	CI	**74.5**	**85.9**	**78.9**	**78.7**	**72.6**	**79.1**	**79.1**	**72.6**	**79.7**	**80.3**	**13.1**

(a)   (b)   (c)
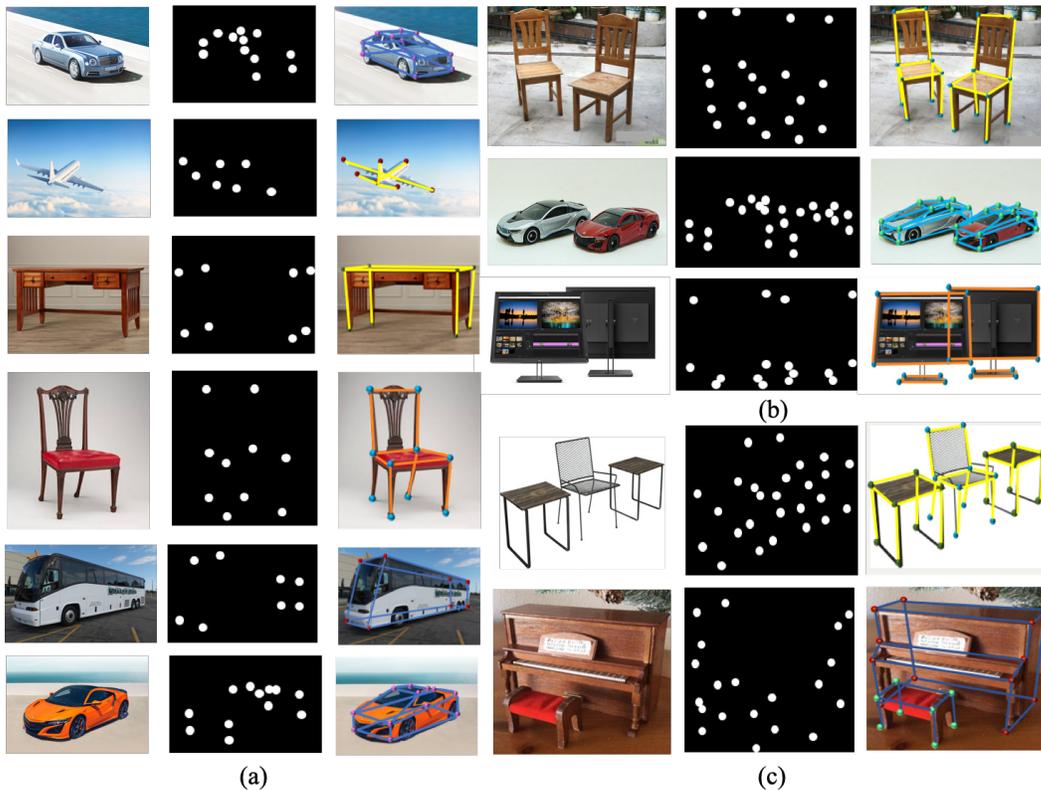
Figure 6: Examples of the experimental results of our final KLPNet. For each panel (a)-(c), the columns from left to right illustrate input image, category-implicit heatmap, and link recovery based on the localized keypoints on the image, respectively. (a) results with respect to a single object; (b) results with respect to multiple single-class objects (from top to bottom: cases for no occlusion, slight occlusion and normal occlusion); (c) results with respect to multiple multi-class objects (from top to bottom: cases for no occlusion and slight occlusion).

an innovative strategy capable of handling occlusion issues; 4) KLPNet, the first end-to-end category-implicit keypoint and link prediction network. The conducted extensive experiments demonstrated both the robustness of our proposed KLPNet, proving the effectiveness of our proposed multi-stage architecture, and meanwhile, showing the state-of-the-art performance on the three publicly available datasets.

References

[1] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.

[2] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer*

vision and pattern recognition, pages 7291–7299, 2017.

[3] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018.

[4] Shubham Goel, Angjoo Kanazawa, and Jitendra Malik. Shape and viewpoint without keypoints. *arXiv preprint arXiv:2007.10982*, 2020.

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

[6] Thomas N Kipf and Max Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

[7] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. Pifpaf: Composite fields for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11977–11986, 2019.

[8] Irene Li, Alexander Fabbri, Swapnil Hingmire, and Dragomir Radev. R-vgae: Relational-variational graph autoencoder for unsupervised prerequisite chain learning. *arXiv preprint arXiv:2004.10610*, 2020.

[9] Wenbo Li, Zhicheng Wang, Binyi Yin, Qixiang Peng, Yuming Du, Tianzi Xiao, Gang Yu, Hongtao Lu, Yichen Wei, and Jian Sun. Rethinking on multi-stage networks for human pose estimation. *arXiv preprint arXiv:1901.00148*, 2019.

[10] 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.

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[12] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8759–8768, 2018.

[13] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Posefix: Model-agnostic general human pose refinement network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7773–7781, 2019.

[14] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in neural information processing systems*, pages 2277–2287, 2017.

[15] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016.

[16] Yan Pang, Jake Christenson, Feng Jiang, Tim Lei, Remy Rhoades, Drew Kern, John A Thompson, and Chao Liu. Automatic detection and quantification of hand movements toward development of an objective assessment of tremor and bradykinesia in parkinson's disease. *Journal of Neuroscience Methods*, 333:108576, 2020.

[17] George Papandreou, Tyler Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, and Kevin Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 269–286, 2018.

[18] Seonwook Park, Shalini De Mello, Pavlo Molchanov, Umar Iqbal, Otmar Hilliges, and Jan Kautz. Few-shot adaptive gaze estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9368–9377, 2019.

[19] 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.

[20] Muhammad Sualeh and Gon-Woo Kim. Simultaneous localization and mapping in the epoch of semantics: a survey. *International Journal of Control, Automation and Systems*, 17(3):729–742, 2019.

[21] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5693–5703, 2019.

[22] Supasorn Suwajanakorn, Noah Snavely, Jonathan J Tompson, and Mohammad Norouzi. Discovery of latent 3d keypoints via end-to-end geometric reasoning. In *Advances in neural information processing systems*, pages 2059–2070, 2018.

[23] Shubham Tulsiani and Jitendra Malik. Viewpoints and keypoints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1510–1519, 2015.

[24] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, and I-Hau Yeh. Cspnet: A new backbone that can enhance learning capability of cnn. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 390–391, 2020.

[25] Yu Xiang, Wonhui Kim, Wei Chen, Jingwei Ji, Christopher Choy, Hao Su, Roozbeh Mottaghi, Leonidas Guibas, and Silvio Savarese. Objectnet3d: A large scale database for 3d object recognition. In *European Conference on Computer Vision*, pages 160–176. Springer, 2016.

[26] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE winter conference on applications of computer vision*, pages 75–82. IEEE, 2014.

[27] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[28] Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren. Distance-iou loss: Faster and better learning for bounding box regression. In *AAAI*, pages 12993–13000, 2020.

[29] Xingyi Zhou, Arjun Karpur, Chuang Gan, Linjie Luo, and Qixing Huang. Unsupervised domain adaptation for 3d keypoint estimation via view consistency. In *Proceedings of the*

European Conference on Computer Vision (ECCV), pages 137–153, 2018.

[30] Xingyi Zhou, Arjun Karpur, Linjie Luo, and Qixing Huang. Starmap for category-agnostic keypoint and viewpoint estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 318–334, 2018.