KGDet: Keypoint-Guided Fashion Detection

Shenhan Qian[*1, 2] **, Dongze Lian**[*1]**, Binqiang Zhao**[2]**, Tong Liu**[2]
Bohui Zhu[2]**, Hai Li**[3]**, Shenghua Gao**[†1, 4]
[1]ShanghaiTech University [2]Alibaba Group [3]Ant Group
[4]Shanghai Engineering Research Center of Intelligent Vision and Imaging
{qianshh, liandz, gaoshh}@shanghaitech.edu.cn
{binqiang.zhao,bohui.zbh}@alibaba-inc.com
yingmu@taobao.com, tianshu.lh@antgroup.com

Abstract

Locating and classifying clothes, usually referred to as clothing detection, is a fundamental task in fashion analysis. Motivated by the strong structural characteristics of clothes, we pursue a detection method enhanced by clothing keypoints, which is a compact and effective representation of structures. To incorporate the keypoint cues into clothing detection, we design a simple yet effective Keypoint-Guided clothing Detector, named KGDet. Such a detector can fully utilize information provided by keypoints with the following two aspects: i) integrating local features around keypoints to benefit both classification and regression; ii) generating accurate bounding boxes from keypoints. To effectively incorporate local features , two alternative modules are proposed. One is a multi-column keypoint-encoding-based feature aggregation module; the other is a keypoint-selection-based feature aggregation module. With either of the above modules as a bridge, a cascade strategy is introduced to refine detection performance progressively. Thanks to the keypoints, our KGDet obtains superior performance on the DeepFashion2 dataset and the FLD dataset with high efficiency.

Introduction

Recently, fashion image understanding has drawn a lot of attention (Liu et al. 2016a; Ge et al. 2019) due to its wide range of applications, especially in online shopping scenarios. A fundamental problem of fashion image analysis is clothing detection, which is to locate and recognize garments efficiently and accurately. It enables many tasks such as clothing attribute analysis, outfit matching, virtual try-on, *etc*.

There has been much work in the field of general object detection (Ren et al. 2015; Redmon et al. 2016; Cai and Vasconcelos 2018), while we focus on clothing detection. For clothes, they usually have very distinct structural features, characterized by keypoints. As shown in Figure 1, keypoints effectively describe the structure of a garment, thus can help to determine the bounding box and the clothing category. Therefore, we propose to leverage the keypoints for fashion detection.

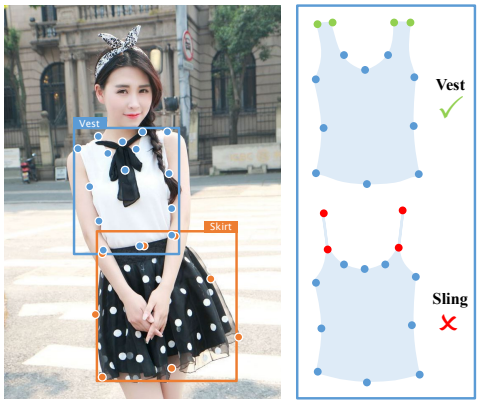

Figure 1: Keypoints effectively describe the structure of a garment, thus can be used to generate high-quality bounding boxes. Meanwhile, the structural information possessed by keypoints can help determine the class of a garment. Here, the keypoints near the shoulders validate that the garment is a vest rather than a sling.

Then a question is raised: how can we incorporate the keypoint cues into clothing detection? At first glance, we can bind clothing detection and keypoint estimation together in a parallel manner, as is done in Mask R-CNN (He et al. 2017), where the extracted feature from ROI-Align is used for object detection and instance segmentation. More recently, Ge et al. (2019) propose a Match R-CNN to integrate clothing detection, landmark regression, segmentation, and clothing retrieval into such a multi-task learning framework. However, in these methods, information of keypoints is not fully utilized but only used to regularize the shared subnetwork to extract features in an implicit way, which results in marginal performance improvement (63.3 vs. 64.0).

Unlike the above solutions, in this paper, we propose a method of serial connection centered on keypoint and design a simple yet effective Keypoint-Guided clothing Detector, named KGDet. Such a detector can make full use of keypoint information with the following two aspects: i) integrating local features around keypoints to benefit both classification and regression; ii) generating accurate bounding boxes from keypoints. Specifically, we design a unified framework

[*]Equal contribution.
[†]Corresponding author.

that first estimates a garment's keypoints, and then explicitly aggregates local features around these keypoints for subsequent refinement. In addition, we directly generate bounding boxes from these predicted keypoints rather than separately regressing them in another branch because the keypoints are real points on a garment, which should be easier to locate than the corner points of bounding box that do not actually exist. Different from previous methods (Ge et al. 2019; Sidnev et al. 2019), where a parallel-connection paradigm is employed, our method is built upon a series-connection strategy between keypoints estimation and clothing detection to make full use of keypoint cues.

Here, a key component is the design of the feature aggregation module for local feature integration around keypoints. An ideal one should satisfy two requirements: i) as the definitions of keypoints vary across categories of clothes, and all the keypoints of all classes are estimated before classification, the module needs to select a subset of keypoints which are valid for a garment; ii) given the selected keypoints, aggregation of features around them is required. For the first requirement, we propose two solutions: i) a multi-column keypoint-encoding-based feature aggregation module that leverages a network to automatically do the information extraction; ii) a keypoints-selection-based feature aggregation module, which selects a part of keypoints according to their regressed confidence scores. As to the second requirement, we handle it with the deformable convolution (Dai et al. 2017), which extracts and fuses features from different positions guided by a learnable deformable offset. Here, we use the offset of selected keypoints as the deformable kernel.

Under ideal circumstances, the more accurate the keypoints are estimated, the better the clothing detection performance is. Based on this hypothesis, we further propose a cascade strategy to refine the predicted keypoints and clothing detection performance repeatedly. Then the above feature aggregation modules can be regarded as a bridge between two cascaded stages.

Our main contributions are summarized as follows:

- we empirically show that keypoints are important cues to help improve the performance of clothing detection and further design a simple yet effective KGDet model that incorporates keypoint cues into clothing detection;
- we provide two alternative modules to aggregate features around keypoints: i) a multi-column keypoint-encoding-based feature aggregation module; ii) a keypoint-selection-based feature aggregation module;
- extensive experiments validate the effectiveness of our method as well as the positive correlation between clothing detection and keypoint estimation. The proposed KGDet achieves superior performance on the DeepFashion2 dataset and FLD dataset with high efficiency.

Related Work

Fashion Image Understanding

The understanding of fashion images is of great commercial value and has inspired much research work. In a survey paper (Cheng et al. 2020), computer vision applications on fashion are categorized into four research topics: fashion detection [1] (Liu et al. 2016b; Yan et al. 2017; Ge et al. 2019), fashion analysis (Chen, Gallagher, and Girod 2012; Chen et al. 2015; Han et al. 2017; Hidayati et al. 2017; Caba Heilbron et al. 2019), fashion synthesis (Yoo et al. 2016; Dong et al. 2019), and fashion recommendation (Song et al. 2019; Wang, Wu, and Zhong 2019; Vasileva et al. 2018). Fashion detection and analysis belongs to the understanding of fashion images, and fashion synthesis and recommendation rely on image understanding.

Early methods for fashion image understanding depend on hand-crafted features (Liu et al. 2012; Kiapour et al. 2014). Recently, with the help of CNNs and large-scale fashion datasets, more and more deep-learning-based applications burst out. For instance, the DeepFashion (Liu et al. 2016a) dataset enables clothing recognition, attribute analysis, and clothing retrieval in simple scenes. Later, The DeepFashion2 dataset (Ge et al. 2019) adds to the difficulty of these problems because it contains multiple garments in a single image and defines finer fashion landmarks for different clothing categories. Our study focuses on multi-class and multi-instance clothing detection since it is closer to the realistic scenario of fashion image understanding, either for customers or businesses.

Fashion Detection and Landmark Detection

Although clothing detection is the foundation of fashion image understanding, little research work is devoted to it specifically. For example, the Match R-CNN (Ge et al. 2019) is adapted from the region-proposal based detector Mask R-CNN (He et al. 2017) with extra branches for fashion landmark estimation and fashion retrieval; the DeepMark (Sidnev et al. 2019) model is a fashion-version of the anchor-free detector CenterNet (Zhou, Wang, and Krähenbühl 2019). In contrast, based on the essential properties of clothing, we design a new method that is guided by fashion landmarks to detect garments in particular.

Fashion landmark detection is to estimate the keypoints (*e.g.,* collar, hemline and waistline) on a garment (Liu et al. 2016b; Yan et al. 2017; Ge et al. 2019; Li et al. 2019; Lee et al. 2019). Liu et al. (2016b) introduce this task and collect a fashion landmark dataset (FLD). Further, a Deep Fashion Alignment (DFA) model is proposed. Wang et al. (2018) utilize a knowledge-guided fashion network to solve the fashion landmark detection problem. Later, Ge et al. (2019) brings the difficulty of this task to a new level with finer keypoint annotations in the DeepFashion2 dataset. Yu et al. (2019) and Lee et al. (2019) use layout-graph reasoning and a global-local embedding module for fashion landmark detection, respectively. All these methods rely heavily on clothing detection, but few of them look into the correlation between fashion landmarks and clothing detection. In this work, we prove that fashion landmarks is beneficial to clothing detection.

[1] In this paper, we do not strictly distinguish between fashion detection and clothing detection, fashion landmark and clothing keypoint.

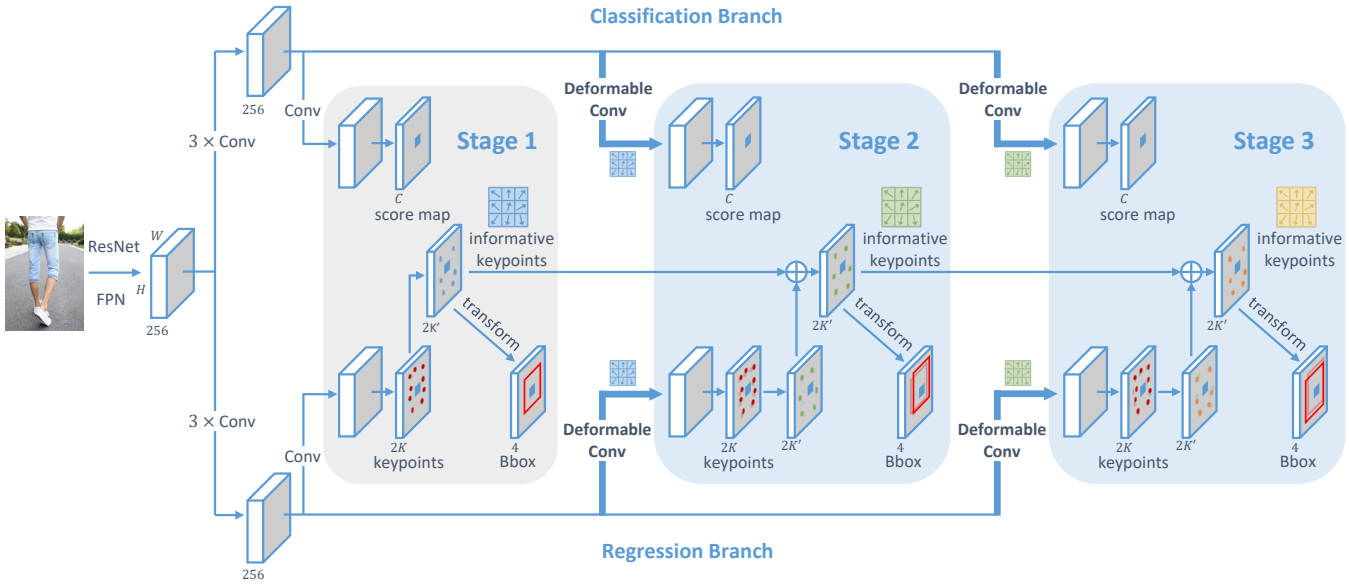

Figure 2: The architecture of the KGDet network, which consists of an initial stage and two refinement stages. Each stage contains two branches: a classification branch and a regression branch. The blue arrow refers to the convolution operation unless otherwise specified. For the depth of feature maps, C refers to the number of classes, K and K' refer to the number of keypoints and informative keypoints, respectively.

Our Approach

Given an input image, the model outputs a bounding box and the corresponding class, and keypoints of each garment. As shown in Figure 2, the KGDet network consists of an initial stage and two refinement stages. In each stage, there is a classification branch and a regression branch. On the classification branch, we predict a score map for each clothing class. On the regression branch, we firstly regress all the keypoints, then compress them into a compact set of points (we call them informative keypoints, which can be viewed as a representative subset of the keypoints). These informative keypoints are used to generate bounding boxes through a transformation. Finally, we utilize keypoint-guided feature aggregation module as a bridge between stages to refine the clothing detection performance gradually.

KGDet Network

Backbone. We use the ResNet (He et al. 2016) along with the Feature Pyramid Network (FPN) (Lin et al. 2017a) as our backbone. According to the training sample assignment strategy of RepPoints (Yang et al. 2019), we find that 99.4% of the garments in the DeepFashion2 dataset (Ge et al. 2019) are assigned to P5-P7 levels of FPN. Therefore, we reduce the computation burden of our model by leaving out the feature maps P3 and P4. The specific statistics can be found in supplementary material.

Before feeding feature maps from the FPN into all three stages, we use three 3×3 convolution layers to decouple the feature maps for classification and those for regression. The difference between an initial stage and a refinement stage is that we use a deformable convolution (Dai et al. 2017) at the beginning of a refinement stage with the informative keypoints serving as the deformable offset.

Classification. We represent an object by its center point, just like other anchor-free detectors (Zhang et al. 2018; Tian et al. 2019; Yang et al. 2019). We predict a score map $\mathbf{C} \in \mathbb{R}^{H \times W \times C}$, where H and W are the height and width of the feature map, and C is the number of categories. A high activation point on the score map $\mathbf{C}$ indicates a candidate object center point of a particular class.

Specifically, on the classification branch, we forward the feature maps for classification through a 3×3 convolution and a 1×1 convolution to generate the score map $\mathbf{C}$. For ground-truth assignment, we align the center point of each ground-truth bounding box onto the score map, and set the k-nearest grids around the point as positive samples ($k = 9$ by default). To train this classification branch, we use a focal loss (Lin et al. 2017b) L_{cls} with $\gamma = 2.0$ and $\alpha = 0.25$.

Keypoint Regression. Similar to RepPoints (Yang et al. 2019), we represent a keypoint with its offset vector from the center of the object, and directly regress the offset values instead of using the heatmap representation.

Specifically, on the regression branch, we feed the features for regression into a 3×3 convolution and a 1×1 convolution to predict the offset of keypoints $\Delta\mathbf{K} \in \mathbb{R}^{H \times W \times 2K}$, where H and W are the height and width of the feature map, and K is the total number of keypoints of all categories. The keypoint offset vector at location (i, j) is

$$\Delta\mathbf{K}_{i,j} = \left[\Delta x_k^1, \Delta y_k^1, \ldots, \Delta x_k^K, \Delta y_k^K\right]^T \in \mathbb{R}^{2K},$$

which encodes the offset values from the location (i, j) to all the K keypoints in x-axis and y-axis direction. Then we

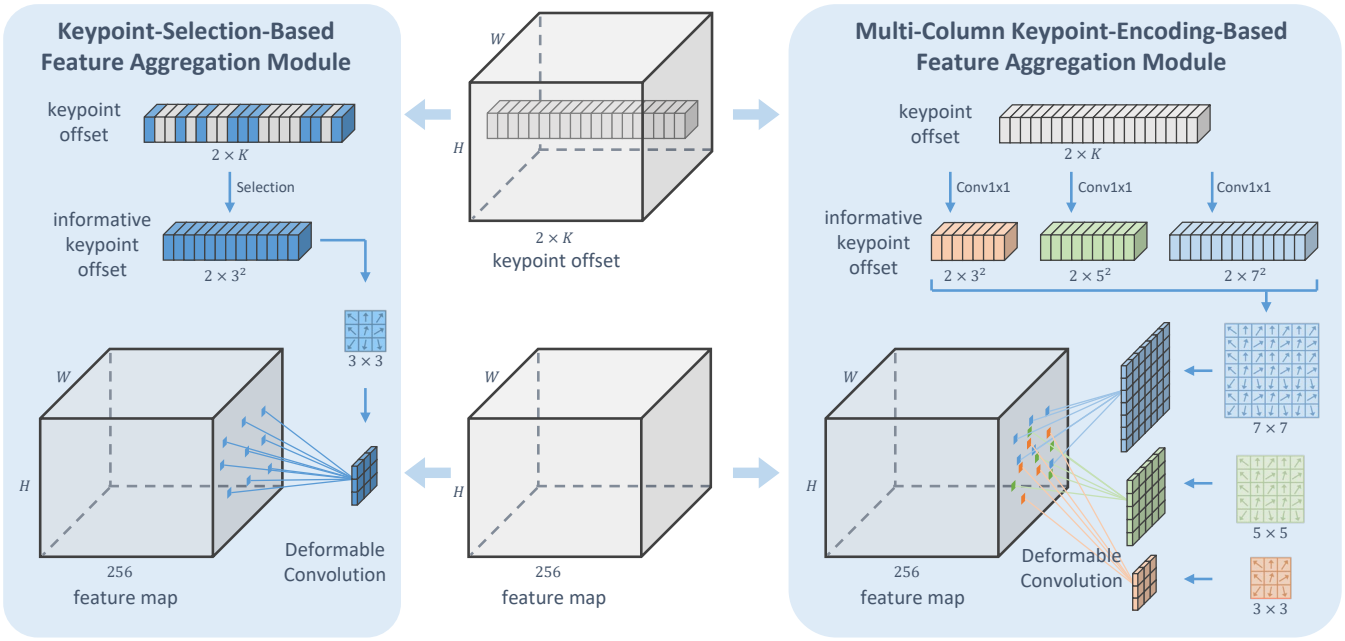

Figure 3: Keypoint-Guided Feature Aggregation.

have the offset vector of the p-th keypoint

$$\Delta \mathbf{K}^p_{i,j} = [\Delta x^p_k, \Delta y^p_k]^T .$$

Finally, the predicted coordinate of the p-th keypoint at location (i, j) is

$$\mathbf{K}^p_{i,j} = [(\Delta x^p_k + i) \times s, (\Delta y^p_k + j) \times s]^T ,$$

where s is the stride of the feature map.

In the training stage, we use the smooth l_1 distance between the predicted keypoint coordinate $\mathbf{K}_{i,j}$ and the ground-truth keypoint coordinate $\widehat{\mathbf{K}}_{i,j}$ at locations of positive samples as our loss function L_{kp}. Note that for each garment, only a subset of keypoints occur in an image. Therefore, we only count the loss of these keypoints that occur.

Informative Keypoints. One of the vital element of our method is the informative keypoints. They serve as a representative subset of all the predicted keypoints. We use them not only to generate bounding boxes but also to aggregate features for succeeding refinement. The reason we use the informative keypoints rather than all the keypoints is that the total number of keypoints can be as large as 294 (*e.g.*, DeepFashion2 (Ge et al. 2019)), and for each garment, only a small proportion of keypoints occur in view.

To obtain the informative keypoints, we propose two solutions, including keypoint-encoding and keypoint-selection. Keypoint-encoding is to generate the informative keypoints $\Delta \mathbf{K}'$ by feeding the predicted offset of keypoints $\Delta \mathbf{K}$ through a 1×1 convolution. Keypoint-selection is to select a subset of keypoints according to a particular standard.

From Informative Keypoints to the Bounding Box. Given the fact that the keypoints of a garment align well with the spatial span of the garment (refer to Figure 1), we generate the bounding box by a transformation from a set of representative keypoints, *i.e.*, the informative keypoints $\mathbf{K}'$. We adapt the transformation trick from RepPoints (Yang et al. 2019) detector, where the mean value and the standard deviation of a point set are used as the center point and the scale of the rectangle box. Two globally shared multipliers λ_x and λ_y are learned to adjust the scale of the box. Having the predicted offset of informative keypoints $\Delta \mathbf{K}' \in \mathbb{R}^{H \times W \times 2K'}$, we can get the offset vector at location (i, j) by

$$\Delta \mathbf{K}'_{i,j} = \left[\Delta x^1_{k'}, \Delta y^1_{k'}, \ldots \Delta x^{K'}_{k'}, \Delta y^{K'}_{k'}\right]^T \in \mathbb{R}^{2K'},$$

where K' is the number of informative keypoints. Then the bounding box is

$$\mathbf{B}_{i,j} = \begin{bmatrix} x_b \\ y_b \\ w_b \\ h_b \end{bmatrix} = \begin{bmatrix} i + \bar{\Delta x}_{k'} \\ j + \bar{\Delta y}_{k'} \\ \lambda_x \cdot s_{\Delta x_{k'}} \\ \lambda_y \cdot s_{\Delta y_{k'}} \end{bmatrix},$$

where $\bar{\Delta x}_{k'}$ and $\bar{\Delta y}_{k'}$ are the mean values of all the informative keypoints, while $s_{\Delta x_{k'}}$ and $s_{\Delta y_{k'}}$ are the standard deviations.

When train the model, we choose the smooth l_1 distance between the predicted bounding box $\mathbf{B}_{i,j}$ and the ground-truth bounding box $\widehat{\mathbf{B}}_{i,j}$ at locations of positive samples as the loss function L_{box} and minimize it.

Keypoint-Guided Feature Aggregation

It is well stated in previous work (He et al. 2017; Wang et al. 2019; Chen et al. 2019) that the alignment of features profoundly influences the performance of a detector.

Figure 4: Visualization of our results on DeepFashion2 (Ge et al. 2019). Our cascaded keypoint-guided feature aggregation network can accurately detect clothing and its keypoints in various scenarios.

We design a keypoint-guided feature aggregation paradigm that sparsely samples features from the locations of informative keypoints. This operation collects more information compared to methods that only use features at the center point of an object. Meanwhile, it avoids gathering background information in region-proposal-based methods (He et al. 2017) because keypoints always lie on the object. Moreover, the keypoint-based informative keypoints reveal a stronger ability of information extraction in contrast with the points learned by the network itself (Yang et al. 2019), which will be shown in our experiments.

Corresponding to the two approaches to obtain informative keypoints, we propose two modules to perform feature aggregation with the aid of deformable convolution (Dai et al. 2017): i) the multi-column keypoint-encoding based feature aggregation module; ii) the reliable-keypoint-selection based feature aggregation module.

Multi-Column Keypoint-Encoding Based Feature Aggregation Module. For feature aggregation with the keypoint-encoding approach, a simple practice is to use a 1×1 convolution to compress K keypoints into K' aggregations points, then conduct deformable convolution with the informative keypoints as the deformation offset. To enhance the expressive ability of the aggregated features, we use three parallel 1×1 convolutions to generate three different sets of informative keypoints. Then we aggregate features with three separate deformable convolutions on each path (Figure 3 right). In the end, the aggregated features on all paths are concatenated before fed forward. The encoded offsets are learned automatically through the back-propagation of the network. This process is depicted in Figure 3 (right).

Reliable-Keypoints-Selection Based Feature Aggregation Module. Another intuitive way to aggregate features is by keypoint-selection. To build a criterion for point selection, we use the ground-truth label for the visibility of each keypoint to train a classifier, predicting whether this keypoint appears in view. As is shown in Figure 3 (left), the predicted keypoint offset $\mathbf{K}$ contains K offset vectors of keypoints at each spatial position. Then, we collect n keypoints whose visibility scores are among the top-n ($n = 9$ by default) as the informative keypoints. Finally, we perform the deformable convolution with the selected keypoint offset. These points are expected to be reliable sources to aggregate features and to help locate the boundary of the garment.

The Cascade Strategy. We adopt the cascade strategy, which has been proven effective in object detection (Cai and Vasconcelos 2018). Our keypoint-guided feature aggregation strategy serves as a bridge between every two stages, enhancing the feature maps for the later stage. Note that the score map of each stage is separately regressed, while the offset of points is calculated the accumulated offset of all the previous stages.

Network Training

Loss Function. We apply supervision on the output of all three stages, so the complete loss of our KGDet is

$$L = \sum_{t=1}^{3} L_{\text{cls}}^{(t)} + \lambda_1 L_{\text{kp}}^{(t)} + \lambda_2 L_{\text{box}}^{(t)}, \tag{1}$$

where t is the index of stage. We empirically set $\lambda_1 = 0.1$ and $\lambda_2 = 1$ to balance different loss terms.

Training Details. We input images with resolution no larger than 1333×800. We train our network with learning rate $5e^{-3}$, momentum 0.9, weight decay $1e^{-4}$, batch size 8 with 4 NVIDIA P40 GPUs, and the SGD optimizer is employed to train the whole network. We only use randomly horizontal flip as data augmentation.

method	keypoint	backbone	AP_{box}	AP^{50}_{box}	AP^{75}_{box}	param.	FLOPs	time (ms)
Match R-CNN (Ge et al. 2019)		ResNet-50-FPN	63.8	78.9	74.5	-	-	-
RetinaNet (Lin et al. 2017b)		ResNet-50-FPN	63.2	79.0	73.6	36.35 M	209.67 G	48.5
FCOS (Tian et al. 2019)		ResNet-50-FPN	64.1	80.2	74.7	**31.87 M**	197.21 G	**38.3**
RepPoints (Yang et al. 2019)		ResNet-50-FPN	63.3	79.7	74.0	36.61 M	189.79 G	55.2
RepPoints-Kp	✓	ResNet-50-FPN	64.0	80.1	74.9	38.09 M	208.84 G	67.6
Ours	✓	ResNet-50-FPN	**67.9**	**80.9**	**75.0**	59.13 M	**107.50 G**	44.6
RetinaNet (Lin et al. 2017b)		ResNet101-FPN	65.7	81.1	75.3	55.35 M	285.74 G	62.1
FCOS (Tian et al. 2019)		ResNet101-FPN	65.9	81.7	75.7	**50.81 M**	273.28 G	**52.6**
RepPoints (Yang et al. 2019)		ResNet101-FPN	65.4	80.8	75.2	55.60 M	265.86 G	69.0
RepPoints-Kp	✓	ResNet-101-FPN	66.0	81.3	75.5	57.08 M	284.91 G	81.3
Ours	✓	ResNet-101-FPN	**68.4**	**82.3**	**76.7**	78.13 M	**183.57 G**	58.5

Table 1: Clothing detection performance comparison of state-of-the-art detectors on the DeepFashion2 (Ge et al. 2019) Dataset.

Inference Details. Given an image with the same resolution as in the training stage, we forward it through our trained network, and obtain the predicted bounding boxes, class labels, and keypoints. We use the same NMS operation as Mask R-CNN (He et al. 2017) for bounding box post-processing. The keypoints at the corresponding position is bundled with the bounding box to go through the same post-processing operation, which means if the bounding box is removed, then the corresponding keypoints at this position is also moved out.

Experimental Setup

Datasets

We evaluate the proposed method on the DeepFashion2 (Ge et al. 2019) and Fashion Landmark Detection (FLD) (Liu et al. 2016b) dataset .

DeepFashion2 (Ge et al. 2019) is a large-scale fashion dataset containing four benchmarks (clothing detection, keypoint estimation, segmentation and commercial-consumer clothes retrieval), with 294 class-relevant keypoints defined for 13 categories of clothes. Since only a subset of the dataset is released (192K images for training, 32K for validation, and 63K for test), our experiments are conducted on this publicly available portion.

FLD (Liu et al. 2016b) defines 8 keypoints for 3 main types of clothes. There are 83K images for training, 19K for validation, and 19K for test.

More details about datasets (distribution of categories and object sizes) are shown in the supplementary materials.

Evaluation Metrics

For clothing detection, the metrics are AP^{50}_{box}, AP^{75}_{box}, and AP_{box}. The first two are the averaged precision under different thresholds of the Intersection over Union (IoU). The last one is the mean value of the averaged precision across ten thresholds.

For fashion landmark detection, the metrics include AP^{50}_{kp}, AP^{75}_{kp}, and AP_{kp}, which are defined in the same way as clothing detection except that we use the Object Keypoint Similarity (OKS) as the threshold.

Baselines

We select varied types of baselines (with or without region-proposal, anchor-based or anchor-free):

Match R-CNN (Ge et al. 2019) is proposed along with the DeepFashion2 dataset. It is extended with extra multi-task branches that perform keypoint estimation and clothing retrieval from *Mask R-CNN* (He et al. 2017) which is based on region-proposal. *RetinaNet* (Lin et al. 2017b) is a single-stage detector which regresses bounding boxes from prior anchors without region-proposal. *FCOS* (Tian et al. 2019) directly regresses bounding boxes from center point features, thus regarded as an anchor-free method. *RepPoints* (Yang et al. 2019) is also an anchor-free detector. It models an object with learned representative points, then extracts features from these points with the deformable convolution (Dai et al. 2017). *RepPoints-Kp* is an extension of *RepPoints* (Yang et al. 2019) implemented by ourselves. It has a keypoint regression head parallel of the box head.

Evaluation Results: Improvement, Correlation and Extension

In this section, we firstly answer the question whether keypoints are beneficial to clothing detection. Based on this, we explore the correlation between clothing detection and keypoint estimation. Since we design two alternative modules for feature aggregation, we compare the performance of them. Finally, we train our KGDet model on the FLD (Liu et al. 2016b) dataset to demonstrate its effectiveness.

Can Keypoints Help Improve Clothing Detection?

To answer the question of whether keypoints can help improve clothing detection, we should first figure out how to integrate keypoints into a detection model. A simple solution is to add an separate keypoint regression branch, which is among our baselines (RepPoints-Kp). Another one is our KGDet network which deeply incorporates keypoints into the process of clothing detection. In Table 1, we compare the clothing detection performance of the two and other recent detectors. Firstly, we can see that our simple solution (RepPoints-Kp) benifits from the additional keypoint supervision and has a performance gain of about 0.7% compared to the vanilla RepPoints (Yang et al. 2019) model.

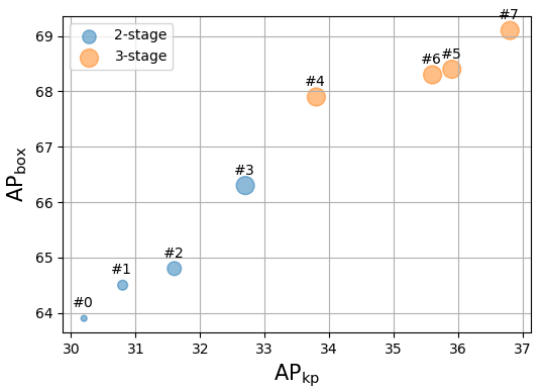

Figure 5: Correlation between clothing detection and keypoint estimation performance on the DeepFashion2 (Ge et al. 2019) Dataset. Each point corresponds to a model (for details see Table 4), whose size is proportional to the number of informative keypoints.

This validates that keypoints do help with clothing detection, although the improvement is marginal. While, our KGDet model outperforms RepPoints-Kp as well as other baseline models [2] by a large margin. This validates the superiority of our keypoint-guided clothing detection design.

As shown in Figure 4, our KGDet network can accurately detect garments and their keypoints in various scenarios. As to the failure cases, in (h), (i), and (j), outwears and shirts are mistaken for tops because the buttons and zips are either vague or occluded. The dress in (l) is confused with a skirt, and the skirt in (m) is wrongly recognized as a part of the shirt, both caused by complex outfit combinations.

Correlation Between Keypoint Estimation and Clothing Detection

Although keypoint estimation is not our goal, it is a vital intermediate task that boosts our detector. It is worth figuring out the correlation between the performance of the two tasks in our model. For simplicity and efficiency, we directly regress keypoints in the form of offset from an object's center points. But, as is stated by previous researches (Sun et al. 2018; Nie et al. 2019), it is hard to precisely regress a large offset with the convolutional neural network. Therefore, we strengthen our model by using more informative keypoints, and refine the predictions in a cascaded way. From Figure 5, we get two observations: i) the performance of clothing detection is positively relevant to that of keypoint estimation; ii) more stages and informative keypoints are beneficial to both tasks. Note that if we disable the supervision of keypoints, the AP_{box} of model #4 drops for 1.2%.

Impact of Different Design Modules

Here we show the comparison of the two feature aggregation modules in Table 2. For our implementation, keypoint-encoding is superior to keypoint-selection.

[2]The performance of Match R-CNN (Ge et al. 2019) is reported in https://github.com/switchablenorms/DeepFashion2.

agg. module	AP_{box}	AP^{50}_{box}	AP^{75}_{box}	AP_{kp}	AP^{50}_{kp}	AP^{75}_{kp}
kp. encoding	**65.8**	**77.9**	**72.1**	**31.7**	**65.4**	**25.3**
kp. selection	64.1	75.3	70.5	29.5	62.9	23.3

Table 2: Comparison of the two proposed feature aggregation modules on the Deepfashion2 dataset with ResNet-50-FPN as the backbone, and features aggregated from 9 points.

Extension to the FLD Dataset

In Table 3, we compare our method with baseline models on the FLD (Liu et al. 2016b) dataset, whose definition of categories and keypoints are simpler. Our method takes the lead by the AP and AP_{75} metrics, indicating higher localization precision, which owes to the keypoint supervision.

method	AP_{box}	AP^{50}_{box}	AP^{75}_{box}
RetinaNet (Lin et al. 2017b)	69.3	90.4	77.8
FCOS (Tian et al. 2019)	68.7	89.9	78.0
RepPoints (Yang et al. 2019)	69.1	**91.3**	78.2
Ours	**70.8**	90.2	**79.0**

Table 3: Performance of clothing detection on the FLD (Liu et al. 2016b) dataset. We use ResNet-50-FPN as the backbone for all the models below.

Conclusion

In this paper, we leverage keypoint cues to improve clothing detection, and propose a KGDet that can integrate the local feature from keypoints for classification and generate accurate bounding box from keypoints for regression. To better gather representative keypoints to help clothing detection, we design two alternative modules, which are the multi-column-keypoint-encoding-based feature aggregation module and the reliable-keypoint-selection-based feature aggregation module, respectively. Extensive experiments show the effect of keypoints to improve clothing detection, and the proposed KGDet achieves superior performance on the DeepFashion2 dataset and FLD dataset with high efficiency.

ID	#pts.	#stages	backbone	FPN levels	$loss_{kp}$	flip	AP_{kp}	AP_{box}
#0	9	2	ResNet-50	$P_5 - P_7$	✓		30.2	63.9
#1	25	2	ResNet-50	$P_5 - P_7$	✓		30.8	64.5
#2	49	2	ResNet-50	$P_5 - P_7$	✓		31.6	65.2
#3	83	2	ResNet-50	$P_5 - P_7$	✓		32.7	66.3
#4	83	3	ResNet-50	$P_5 - P_7$	✓		33.8	67.9
#5	83	3	ResNet-101	$P_5 - P_7$	✓		35.9	68.4
#6	83	3	ResNet-50	$P_5 - P_7$	✓	✓	35.6	68.3
#7	83	3	ResNet-50	$P_4 - P_7$	✓	✓	36.8	69.1

Table 4: Clothing detection performance improves with keypoint estimation. "#pts": the number of informative points; "#stages": the number of stages; "FPN levels": the levels of FPN's feature maps as input; "flip": feeding in an input image and its horizontal mirror into the model, and fusing the output when training and testing.

Acknowledgments

The work was supported by National Key R&D Program of China (2018AAA0100704), NSFC #61932020, Science and Technology Commission of Shanghai Municipality (Grant No. 20ZR1436000) and Alibaba Innovation Research.

References

Caba Heilbron, F.; Pepik, B.; Barzelay, Z.; and Donoser, M. 2019. Clothing Recognition in the Wild using the Amazon Catalog. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 0–0.

Cai, Z.; and Vasconcelos, N. 2018. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6154–6162.

Chen, H.; Gallagher, A.; and Girod, B. 2012. Describing clothing by semantic attributes. In *European conference on computer vision*, 609–623. Springer.

Chen, Q.; Huang, J.; Feris, R.; Brown, L. M.; Dong, J.; and Yan, S. 2015. Deep domain adaptation for describing people based on fine-grained clothing attributes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5315–5324.

Chen, Y.; Han, C.; Wang, N.; and Zhang, Z. 2019. Revisiting feature alignment for one-stage object detection. *arXiv preprint arXiv:1908.01570* .

Cheng, W.-H.; Song, S.; Chen, C.-Y.; Hidayati, S. C.; and Liu, J. 2020. Fashion Meets Computer Vision: A Survey. *arXiv preprint arXiv:2003.13988* .

Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; and Wei, Y. 2017. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 764–773.

Dong, H.; Liang, X.; Shen, X.; Wu, B.; Chen, B.-C.; and Yin, J. 2019. FW-GAN: Flow-navigated Warping GAN for Video Virtual Try-on. In *Proceedings of the IEEE International Conference on Computer Vision*, 1161–1170.

Ge, Y.; Zhang, R.; Wang, X.; Tang, X.; and Luo, P. 2019. Deepfashion2: A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5337–5345.

Han, X.; Wu, Z.; Huang, P. X.; Zhang, X.; Zhu, M.; Li, Y.; Zhao, Y.; and Davis, L. S. 2017. Automatic spatially-aware fashion concept discovery. In *Proceedings of the IEEE International Conference on Computer Vision*, 1463–1471.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hidayati, S. C.; You, C.-W.; Cheng, W.-H.; and Hua, K.-L. 2017. Learning and recognition of clothing genres from full-body images. *IEEE transactions on cybernetics* 48(5): 1647–1659.

Kiapour, M. H.; Yamaguchi, K.; Berg, A. C.; and Berg, T. L. 2014. Hipster wars: Discovering elements of fashion styles. In *European conference on computer vision*, 472–488. Springer.

Lee, S.; Oh, S.; Jung, C.; and Kim, C. 2019. A Global-local Embedding Module for Fashion Landmark Detection. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 0–0.

Li, Y.; Tang, S.; Ye, Y.; and Ma, J. 2019. Spatial-Aware Non-Local Attention for Fashion Landmark Detection. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, 820–825. IEEE.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017a. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017b. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Liu, S.; Song, Z.; Liu, G.; Xu, C.; Lu, H.; and Yan, S. 2012. Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 3330–3337. IEEE.

Liu, Z.; Luo, P.; Qiu, S.; Wang, X.; and Tang, X. 2016a. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1096–1104.

Liu, Z.; Yan, S.; Luo, P.; Wang, X.; and Tang, X. 2016b. Fashion landmark detection in the wild. In *European Conference on Computer Vision*, 229–245. Springer.

Nie, X.; Feng, J.; Zhang, J.; and Yan, S. 2019. Single-stage multi-person pose machines. In *Proceedings of the IEEE International Conference on Computer Vision*, 6951–6960.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 91–99.

Sidnev, A.; Trushkov, A.; Kazakov, M.; Korolev, I.; and Sorokin, V. 2019. Deepmark: One-shot clothing detection. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 0–0.

Song, X.; Han, X.; Li, Y.; Chen, J.; Xu, X.-S.; and Nie, L. 2019. GP-BPR: Personalized Compatibility Modeling for

Clothing Matching. In *Proceedings of the 27th ACM International Conference on Multimedia*, 320–328.

Sun, X.; Xiao, B.; Wei, F.; Liang, S.; and Wei, Y. 2018. Integral human pose regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 529–545.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 9627–9636.

Vasileva, M. I.; Plummer, B. A.; Dusad, K.; Rajpal, S.; Kumar, R.; and Forsyth, D. 2018. Learning type-aware embeddings for fashion compatibility. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 390–405.

Wang, J.; Chen, K.; Yang, S.; Loy, C. C.; and Lin, D. 2019. Region proposal by guided anchoring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2965–2974.

Wang, W.; Xu, Y.; Shen, J.; and Zhu, S.-C. 2018. Attentive fashion grammar network for fashion landmark detection and clothing category classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4271–4280.

Wang, X.; Wu, B.; and Zhong, Y. 2019. Outfit Compatibility Prediction and Diagnosis with Multi-Layered Comparison Network. In *Proceedings of the 27th ACM International Conference on Multimedia*, 329–337.

Yan, S.; Liu, Z.; Luo, P.; Qiu, S.; Wang, X.; and Tang, X. 2017. Unconstrained fashion landmark detection via hierarchical recurrent transformer networks. In *Proceedings of the 25th ACM international conference on Multimedia*, 172–180.

Yang, Z.; Liu, S.; Hu, H.; Wang, L.; and Lin, S. 2019. Reppoints: Point set representation for object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 9657–9666.

Yoo, D.; Kim, N.; Park, S.; Paek, A. S.; and Kweon, I. S. 2016. Pixel-level domain transfer. In *European Conference on Computer Vision*, 517–532. Springer.

Yu, W.; Liang, X.; Gong, K.; Jiang, C.; Xiao, N.; and Lin, L. 2019. Layout-graph reasoning for fashion landmark detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2937–2945.

Zhang, S.; Wen, L.; Bian, X.; Lei, Z.; and Li, S. Z. 2018. Single-shot refinement neural network for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4203–4212.

Zhou, X.; Wang, D.; and Krähenbühl, P. 2019. Objects as points. *arXiv preprint arXiv:1904.07850* .