

# 基于改进 CenterNet 的托盘定位方法<sup>①</sup>



朱丹平, 朱 明, 周恒森

(中国科学技术大学 信息科学技术学院, 合肥 230026)

通信作者: 朱丹平, E-mail: danpingz@mail.ustc.edu.cn

**摘 要:** 目前, 托盘定位大多采用基于深度神经网络的目标检测算法, 一般使用矩形框进行托盘定位, 托盘中心点定位精度不高, 且无法有效估计托盘水平方向. 针对此问题, 本文提出了基于关键点检测的托盘定位方法, 通过检测托盘正面外轮廓的 4 个角点来定位托盘. 首先, 由于目前没有大规模的托盘数据集, 使用迁移学习的方法, 将 CenterNet 的人体姿态估计引入托盘定位任务. 然后改进关键点分组方法, 并提出关键点回归自适应补偿, 提高关键点检测精度. 在托盘关键点定位的基础上, 提出基于几何约束的托盘中心点计算和托盘水平方向估计方法. 本文方法与原 CenterNet 相比, 托盘关键点定位指标  $AP^{kp}$  从 0.352 提高到 0.728, 托盘中心点定位精度指标  $ALP$  达到 0.946, 并且可以有效估计托盘水平方向, 具有较高的实用价值.

**关键词:** 托盘定位; 目标检测; 关键点检测; 中心点定位; 方向估计; 深度学习

引用格式: 朱丹平, 朱明, 周恒森. 基于改进 CenterNet 的托盘定位方法. 计算机系统应用, 2022, 31(10):303-309. <http://www.c-s-a.org.cn/1003-3254/8728.html>

## Pallet Positioning Method Based on Improved CenterNet

ZHU Dan-Ping, ZHU Ming, ZHOU Heng-Sen

(School of Information Science and Technology, University of Science and Technology of China, Hefei 230026, China)

**Abstract:** Currently, the target detection algorithm based on depth neural network is mostly used for pallet positioning and a rectangular box is generally utilized. The positioning accuracy of the pallet center point is not high enough, and the horizontal direction of the pallet cannot be estimated effectively. To solve this problem, this study proposes a pallet positioning method based on keypoint detection, which locates the pallet by detecting the four corners of the front outer outline. Firstly, due to the shortage of large-scale pallet data sets, the human posture estimation of CenterNet is introduced by transfer learning. Then the keypoint grouping method is improved, and the adaptive compensation is proposed for keypoint regression to improve the keypoint detection accuracy. According to the location of pallet keypoints, a method of pallet center point calculation and pallet horizontal direction estimation based on geometric constraints is proposed. Compared with the original CenterNet, the proposed method raises the positioning index  $AP^{kp}$  of pallet keypoint from 0.352 to 0.728, and the positioning accuracy  $ALP$  of pallet center point to 0.946. Meanwhile, it can effectively estimate the pallet horizontal direction and is of high practical value.

**Key words:** pallet positioning; target detection; keypoint detection; center point positioning; direction estimation; deep learning

## 1 引言

随着我国物流行业的发展, 以及劳动力成本的不  
断上升, 自动化仓储物流近年来不断发展, 自动化运输

设备的需求不断提升. 自动叉车是一种自动驾驶车辆,  
可以完成对货物的自动搬运, 能有效降低人力成本, 被  
广泛应用于自动化物流中. 在仓储物流中, 因为货物的

① 基金项目: 科技创新特区计划 (20-163-14-LZ-001-004-01)

收稿时间: 2022-01-17; 修改时间: 2022-02-17; 采用时间: 2022-02-22; csa 在线出版时间: 2022-06-28

外观存在大小、形状、颜色等明显的差异,所以直接对货物定位非常困难.而托盘用于承载和搬运货物,且具有通用的规格和形状.因此自动叉车一般通过托盘定位来叉取和搬运货物.

托盘定位方法主要分为以下3类:第1类是基于激光雷达传感器的检测方法.赵俊宏等人<sup>[1]</sup>根据2D激光雷达扫描结果分析托盘的距离和姿态.虽然基于激光雷达的方法对光照有比较好的鲁棒性,但是激光雷达成本高昂、不利于大规模普及.第2类是传统图像处理的方法.Cui等人<sup>[2]</sup>使用传统图像处理的方法直接分析图像本身特征,如角点、边缘和颜色等特征来检测托盘.Seelinger等人<sup>[3]</sup>提出了一种利用标签来定位托盘的方法,但存在标签安装和维护的成本,而且在使用过程中容易损坏.武武汉<sup>[4]</sup>使用tof相机和纸标签分别完成了对远距离托盘和近距离托盘的检测定位,但是算法速度较慢,实时性不高,因此难以应用于工业场景.此类方法易受到光照等环境因素影响,鲁棒性较差.第3类是基于深度学习的托盘检测方法.Mohamed等人<sup>[5]</sup>使用了Faster R-CNN<sup>[6]</sup>来训练模型,实现了对托盘的定位.李天健等人<sup>[7]</sup>使用改进的SSD算法<sup>[8]</sup>实现了仓储环境中托盘的定位.叶伟<sup>[9]</sup>首先使用YOLOv4算法<sup>[10]</sup>对托盘进行目标检测,再使用GrabCut分割方法<sup>[11]</sup>提高精度.此类方法准确率高,抗干扰能力强,而且成本远低于激光雷达,适合部署于实际工业场景中.

在实际物流场景中,自动叉车为完成自动叉卸货任务,需要将叉铲对准托盘中心点,将叉铲方向与托盘水平方向保持一致,以保证安全无碰撞地叉取托盘.因此托盘定位任务可分为2部分:(1)托盘中心点定位;(2)托盘水平方向估计.目前基于目标检测的托盘定位算法,一般使用矩形框进行定位,只能检测托盘区域位置,使用目标框中心点对托盘中心点定位精度不高,且无法有效估计托盘水平方向,难以应用于自动叉车系统中.基于此问题,本文提出了一种基于改进CenterNet<sup>[12]</sup>的托盘定位方法.在目标检测的基础上,通过托盘关键点检测,在精准定位托盘中心点的同时,完成对托盘水平方向的估计.

本文的贡献如下:(1)本文使用迁移学习的方法,将人体姿态估计引入托盘定位任务,完成了托盘关键点定位.(2)针对托盘定位任务,本文对CenterNet网络做出2点改进:1)改进关键点分组方法,提出基于托盘关键点位置的区域约束方法;2)提出关键点回归评价

指标与自适应补偿,提高关键点检测精度.(3)在关键点定位的基础上,提出基于几何约束的托盘中心点计算方法,以及托盘水平方向估计方法.(4)使用RGB摄像头拍摄并标注了托盘数据集.

## 2 相关工作

Mohamed等人<sup>[5]</sup>、李天健等人<sup>[7]</sup>与叶伟<sup>[9]</sup>所使用的目标检测算法,都需要人工设计特定长宽比的锚框,以训练出更好的检测器.但是人工设计的锚框难以兼顾所有长宽比的目标,针对此问题,Law等人提出了CornerNet<sup>[13]</sup>,通过检测目标的左上角和右下角关键点来检测目标,不需要人工设计锚框.ExtremeNet<sup>[14]</sup>通过检测目标的顶部、底部、最左侧、最右侧和中心点这5点信息来完成目标检测.以上两种无锚框目标检测算法在关键点检测之后,都需要进行复杂且耗时的后处理步骤,这大大降低了算法的实时性.而Zhou等人<sup>[12]</sup>提出的CenterNet较好解决了这个问题.CenterNet将目标检测简化成为对目标中心点的检测.使用3×3最大池化选取目标中心点,从而避免了NMS等后处理步骤.

### 2.1 CenterNet 目标检测算法

CenterNet可选取3种不同的骨干网络,分别为Hourglass<sup>[15]</sup>、ResNet<sup>[16]</sup>、DLA<sup>[17]</sup>,都是编码-解码结构.其中Hourglass在3种骨干网络中精度最高但预测速度较慢,而使用DLA可以得到较好的精度以及较快的速度,因此选择DLA作为算法的骨干网络.如图1所示,CenterNet网络的头部可分为3个互相独立的部分,分别完成目标框中心点、中心点偏置和目标框宽高的检测.CenterNet使用二维高斯核函数将输入图像映射成为热力图作为训练目标.二维高斯核函数的公式为:

$$Y_{xyc} = \exp\left(-\frac{(x-\tilde{p}_x)^2 + (y-\tilde{p}_y)^2}{2\sigma_p^2}\right) \quad (1)$$

其中, $Y_{xyc}$ 是归一化后的热力图, $(x,y)$ 为热力图中点的坐标, $(\tilde{p}_x,\tilde{p}_y)$ 为目标中心点4倍下采样后的坐标(向下取整), $\sigma_p$ 是自适应标准差<sup>[13]</sup>.CenterNet目标检测任务中损失函数为 $L_{det}$ :

$$L_{det} = L_k + \lambda_{size}L_{size} + \lambda_{off}L_{off} \quad (2)$$

$L_{det}$ 由3部分组成,分别为置信度损失 $L_k$ 、下采样损失 $L_{off}$ 和目标框宽高损失 $L_{size}$ ,超参数 $\lambda_{size}$ 为0.1, $\lambda_{off}$ 为1. $L_k$ 用来训练目标中心点位置.由于正样本远少于负样本数量,为了使模型在训练时更加关注于难检

测的样本,  $L_k$  使用 focal loss<sup>[18]</sup>. 因为输出的热力特征图  
为原图大小的 0.25 倍, 所以中心点位置预测存在离散  
化误差,  $L_{\text{off}}$  用来训练下采样偏置的大小.

$$L_k = \frac{-1}{N} \sum_{\text{xyz}} \begin{cases} (1 - \hat{Y}_{\text{xyz}})^\alpha \log(\hat{Y}_{\text{xyz}}), & \text{if } Y_{\text{xyz}} = 1 \\ (1 - Y_{\text{xyz}})^\beta (\hat{Y}_{\text{xyz}})^\alpha \\ \log(1 - \hat{Y}_{\text{xyz}}), & \text{otherwise} \end{cases} \quad (3)$$

其中,  $Y_{\text{xyz}} = 1$  表示正样本,  $\hat{Y}_{\text{xyz}}$  是目标检测的预测值.  
 $N$  是图像中正样本的数量. 超参数  $\alpha$  和  $\beta$  分别为 2 和 4.

$$L_{\text{off}} = \frac{1}{N} \sum_p \left| \hat{O}_p - \left( \frac{p}{R} - \hat{p} \right) \right| \quad (4)$$

其中,  $\hat{O}_p$  为预测的中心点偏置大小,  $p$  为中心点真实值  
坐标,  $R$  为下采样倍数, 中心点坐标  $p$  下采样  $R$  倍后向下  
取整得到  $\hat{p}$ .

$$L_{\text{size}} = \frac{1}{N} \sum_{k=1}^N |\hat{s}_{pk} - s_k| \quad (5)$$

其中,  $s_k = (w^k, h^k)$  为目标  $k$  的宽高,  $\hat{s}_{pk}$  为宽高估计值.

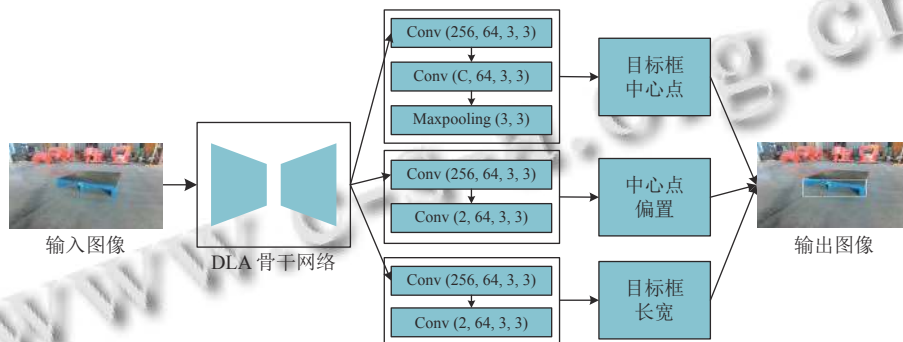


图 1 CenterNet 网络结构图

### 2.2 CenterNet 人体姿态估计

CenterNet 具有良好可扩展性, 可以将 CenterNet  
应用到人体姿势估计任务中. 人体姿势估计的任务要  
求估计出图像中每个人体实例的人体关键点位置. 首  
先直接回归每个关键点与中心点之间的偏移量, 得到  
关键点的初步定位, 使用 L1 损失函数  $L_{\text{reg}}$  进行训练. 然  
后为了细化关键点定位, 类似目标检测, 通过热力特  
征图以及中心点偏置来进行关键点检测, 并以初步定  
位位置为依据进行分组. 关键点检测使用目标检测任  
务中的置信度损失  $L_k$  和偏移损失  $L_{\text{off}}$  进行训练.

$$L_{\text{reg}} = \frac{1}{N} \sum_{k=1}^N \frac{1}{\sum_{i=1}^E M_{ki}} \sum_{j=1}^E |\hat{J}_{kj} - J_{kj}| \times M_{kj} \quad (6)$$

其中,  $E$  为关键点个数,  $M$  为关键点掩码, 对于图像中被  
遮挡等不可见的关键点  $M_{ki} = 0$ , 可见关键点  $M_{ki} = 1$ .  
 $\hat{J}_{kj} = (x_{kj}, y_{kj})$  为第  $k$  个目标的第  $j$  个关键点坐标的偏移  
量. 关键点检测失函数为  $L_{\text{key}} = L_k + L_{\text{off}} + L_{\text{reg}}$ , 得到总  
损失函数  $L_{\text{oss}}$ :

$$L_{\text{oss}} = L_{\text{det}} + L_{\text{key}} \quad (7)$$

在多人姿态检测场景中, 通过关键点检测得到的  
结果无法确定属于哪个人体实例. CenterNet 使用初步

定位结果作为分组依据, 使用 3 项规则进行匹配:

- (1) 所匹配的热力特征图关键点置信度得分大于 0.1;
- (2) 匹配欧氏距离最近的初步定位位置;
- (3) 仅匹配目标框内的关键点.

## 3 算法

### 3.1 算法框架

如图 2 所示, 不同类型托盘外观存在一定差异, 主  
要可分为两柱形托盘和三柱形托盘. 为适应不同类型  
托盘, 选取托盘正面外轮廓的 4 个角点作为关键点. 从  
左上角点开始顺时针将关键点分别编号为 1, 2, 3, 4. 托  
盘定位算法可分为托盘检测、托盘关键点检测以及托  
盘中心点计算与水平方向估计的后处理部分. 算法框  
架如图 3 所示, 其具体流程如算法 1 所示. 算法 1 中  
的所有步骤均在图 3 中标出, 其中本文提出的工作用星  
号标记, 在原算法基础上改进的工作用菱形标记.

#### 算法 1. 托盘定位算法

- 1) 将输入图片通过仿射变换转为  $512 \times 512$  固定大小输入;
- 2) 通过骨干网络生成  $128 \times 128$  高分辨率热力特征图;
- 3) 基于热力特征图进行托盘检测获得托盘目标框;
- 4) 直接回归得到托盘关键点初步定位位置;
- 5) 使用自适应补偿对初步定位位置进行调整;
- 6) 使用热力特征图进行关键点检测;



7) 以托盘目标框和关键点初步定位位置为分组依据, 对步骤 6) 的结果进行关键点分组;

8) 基于托盘关键点定位结果, 计算托盘中心点, 估计托盘的水平方向.

### 3.2 基于关键点位置的区域约束

CenterNet 网络为了对关键点进行分组, 只对目标框内的关键点进行匹配. 由于托盘关键点分布于托盘边缘, 托盘检测轻微的精度误差就可能导致目标框无法包含关键点位置, 严重影响关键点分组结果, 针对此问

题, 本文根据托盘关键点分布特点, 如图 4 所示, 分别为左右两边的关键点设置了两组约束区域.

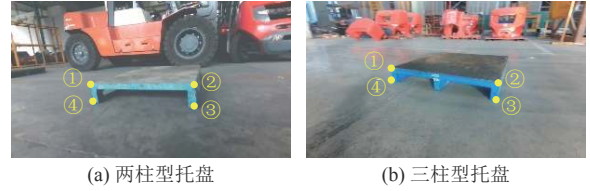


图 2 托盘类型图

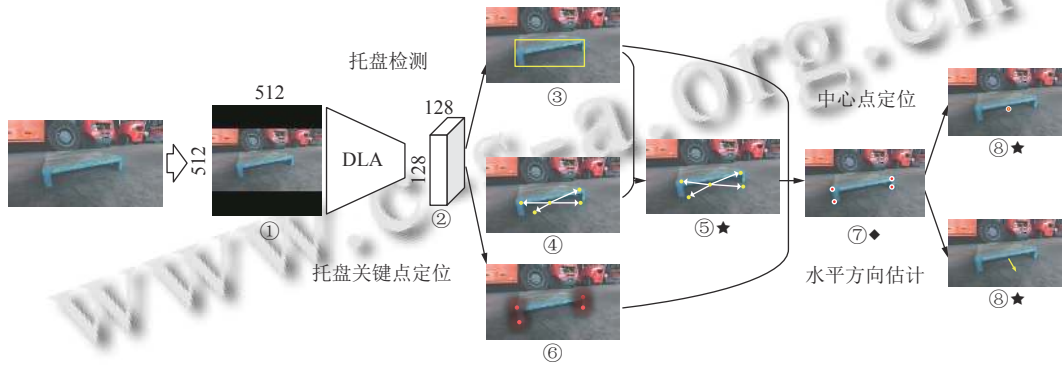


图 3 托盘定位算法图

$$\begin{cases} P_{\text{right}} = \begin{cases} |x - x_m - 0.5w| < 0.15w \\ |y - y_m| < 0.6h \end{cases} \\ P_{\text{left}} = \begin{cases} |x - x_m + 0.5w| < 0.15w \\ |y - y_m| < 0.6h \end{cases} \end{cases} \quad (8)$$

其中,  $P_{\text{left}}$  和  $P_{\text{right}}$  分别为托盘左右两边关键点的约束区域,  $(x_m, y_m)$  为托盘目标框中心点,  $(w, h)$  为托盘目标框宽高,  $(x, y)$  为图像中点的坐标.

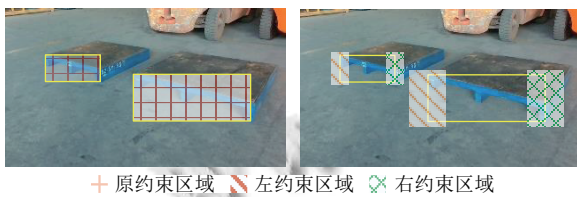


图 4 区域约束对比图

### 3.3 关键点回归自适应补偿

关键点回归结果作为关键点定位的初步结果, 是关键点检测的分组依据. 较差的关键点回归结果, 会直接影响关键点定位的最终精度. 因为关键点分布于托盘外边缘, 所以关键点的最小外接矩形与托盘的目标框相近. 托盘目标框中心点与第  $i$  个关键点的欧式距离为  $D_i^{\text{reg}}$ , 目标框中心点与目标框角点欧式距离  $D_{\text{det}}$ .  $D_{\text{det}}$  反映了目标的大小情况, 对  $D_i^{\text{reg}}$  有一定参考意义.

在  $D_i^{\text{reg}}$  与  $D_{\text{det}}$  数值差距较大时, 认为托盘关键点回归效果较差. 根据二者大小关系得到关键点回归评价指标  $\rho_i^{\text{reg}}$ .

$$\rho_i^{\text{reg}} = 1 - \min\left(1, \left|1 - \frac{D_i^{\text{reg}}}{D_{\text{det}}}\right|\right) \quad (9)$$

其中,  $\rho_i^{\text{reg}}$  为第  $i$  个关键点回归评价,  $\rho_i^{\text{reg}}$  越高代表回归效果越好, 根据  $\rho_i^{\text{reg}}$  使用对应的目标框角点坐标对关键点回归结果进行补偿.

$$P_i^{\text{reg}} = \rho_i^{\text{reg}} P_i^{\text{reg}} + (1 - \rho_i^{\text{reg}}) P_i^{\text{rect}} \quad (10)$$

其中,  $P_i^{\text{reg}} = (x_i, y_i)$  为第  $i$  个托盘关键点回归的坐标. 将目标框角点从左上角按顺时针编号为 1, 2, 3, 4,  $P_i^{\text{rect}} = (x_i^{\text{rect}}, y_i^{\text{rect}})$  为第  $i$  个托盘目标框角点坐标. 当  $D_i^{\text{reg}}$  与  $D_{\text{det}}$  接近时,  $\rho_i^{\text{reg}}$  接近于 1, 关键点回归效果好, 补偿比重小; 当  $D_i^{\text{reg}}$  与  $D_{\text{det}}$  差距较大时,  $\rho_i^{\text{reg}}$  接近于 0, 关键点回归效果差, 补偿比重大.

### 3.4 基于几何约束的托盘中心点计算方法

自动叉车相机与托盘空间关系如图 5 所示, 坐标系  $z$  轴指向相机正前方,  $y$  轴垂直向下. 因为托盘承载货物一般水平放置于地面, 所以设定托盘放置平面与自动叉车相机坐标系  $xz$  平面平行.  $P_i$  为托盘关键点

( $i = 1, 2, 3, 4$ ),  $P_i$ 在自动叉车相机坐标系中的坐标为( $X_i, Y_i, Z_i$ ).  $P_c$ 为托盘中心点,在自动叉车相机坐标系中的坐标为( $X_c, Y_c, Z_c$ ).  $p_i$ 为托盘关键点在归一化平面中的坐标( $x_i, y_i$ ),  $p_c$ 为托盘中心点在归一化平面的坐标( $x_c, y_c$ ). 基于针孔相机模型,本文提出基于几何约束的托盘中心点计算方法,通过托盘关键点计算托盘中心点.

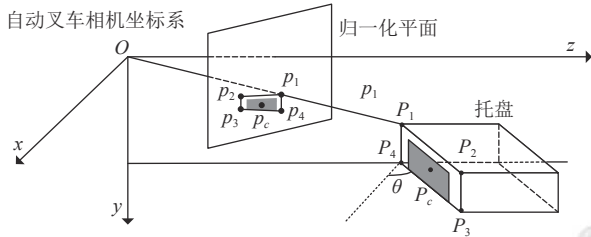


图5 叉车相机与托盘空间关系图

$$\begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} = \frac{1}{Z_i} \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix} \triangleq \frac{1}{Z_i} K P_i \quad (11)$$

其中,  $K$ 为相机内参矩阵. 因为托盘放置平面与自动叉车相机坐标系 $xz$ 平面平行,在自动叉车相机坐标系中 $Z_1 = Z_4, Z_2 = Z_3$ ,根据式(11)可得:

$$\mu = \frac{Z_1}{Z_2} = \frac{y_2 - y_3}{y_1 - y_4} \quad (12)$$

托盘中心点 $P_c$ 与托盘关键点的空间关系为:

$$X_c = 0.5(X_1 + X_2) = 0.5(X_3 + X_4) \quad (13)$$

$$Y_c = 0.5(Y_1 + Y_4) = 0.5(Y_2 + Y_3) \quad (14)$$

$$Z_c = 0.5(Z_1 + Z_2) = 0.5(Z_3 + Z_4) \quad (15)$$

将式(12)–式(15)带入式(11)得到托盘中心点坐标( $x_c, y_c$ ):

$$\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \frac{1}{2(\mu + 1)} \begin{pmatrix} 2x_1 & 2x_2 \\ y_1 + y_4 & y_2 + y_3 \end{pmatrix} \begin{pmatrix} 1 \\ \mu \end{pmatrix} \quad (16)$$

### 3.5 托盘水平方向估计

为了估计托盘水平方向,本文在自动叉车相机坐标系中分析了托盘关键点坐标与水平方向的关系.如图5,定义托盘水平方向的角度 $\theta$ 为托盘正面所在平面 $P_1P_2P_3$ 与坐标系 $x$ 轴正方向所成夹角,设定 $Z_1 < Z_2$ 时 $\theta$ 为正,反之为负.通过托盘关键点 $p_1$ 和 $p_2$ 的 $y$ 坐标之差 $\Delta y = y_1 - y_2$ 来估计托盘水平方向.关键点 $P_1$ 和 $P_2$ 在空间中的关系如式(17),其中 $W$ 为关键点 $P_1$ 和 $P_2$ 的空间距离,也即托盘正面长度.

$$\begin{pmatrix} X_2 \\ Y_2 \\ Z_2 \end{pmatrix} = \begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} + \begin{pmatrix} W \cos \theta \\ 0 \\ W \sin \theta \end{pmatrix} \quad (17)$$

将式(17)带入式(11),得到:

$$\Delta y = \frac{f_y Y_1 W}{Z_1} \left( \frac{\sin \theta}{Z_1 + W \sin \theta} \right) \quad (18)$$

在 $Z_1$ 不变的前提下, $\Delta y$ 对 $\theta$ 求导:

$$\Delta y' = \frac{f_y Y_1 W \cos \theta}{(Z_1 + W \sin \theta)^2} \quad (19)$$

由式(18)和式(19)可知,当 $\theta$ 为0时, $p_1$ 和 $p_2$ 的 $y$ 坐标之差 $\Delta y$ 为0;当 $\theta$ 大于0时, $\Delta y$ 大于0;当 $\theta$ 小于0时, $\Delta y$ 小于0.同一距离下 $\Delta y$ 在 $\theta \in [-0.5\pi, 0.5\pi]$ 的区间内单调递增,因此通过 $\Delta y$ 可以有效估计托盘水平方向.考虑到关键点检测误差,设 $\Delta y \in [-10, 10]$ 时,托盘水平方向正对; $\Delta y < -10$ 时,托盘水平方向左偏; $\Delta y > 10$ 时,托盘水平方向右偏.

## 4 实验分析

### 4.1 数据集

本文采用的数据集为笔者自己采集制作的数据集.笔者使用 $1920 \times 1080$ 分辨率的RGB摄像头拍摄了一个数据集.如图6所示,摄像头安装在自动叉车系统前方的叉齿架上.叉齿架可上升下降,在不同高度拍摄托盘照片.整个数据集拍摄了合力叉车公司2种类型的托盘,在多个场景中采集了时长60分钟的视频.从中选取1251张图片,使用coco-annotator工具标注.将其中80%图片作为训练集,10%用作验证集,10%用作测试集.

### 4.2 实验过程

实验所用的机器配置如表1所示.实验有两个评价指标.第1个是关键点定位精度指标 $AP^{kp}$ ,用以衡量关键点定位的精度, $AP^{kp}$ 越高代表托盘关键点定位越精确.第2个指标是笔者提出的托盘中心点定位指标平均定位精度 $ALP$  (average location precision),通过托盘定位中心点与托盘实际中心点的欧式距离 $D_c$ 和托盘真实框面积 $A_{bbox}$ 得到.

$$ALP = 1 - \frac{1}{N} \sum_{i=1}^N \frac{D_c}{\sqrt{A_{bbox}}} \quad (20)$$

在 $ALP$ 中只考虑检测目标框和真实框IOU大于0.5的样本,式(20)中 $N$ 为样本数量. $ALP$ 越高代表对托盘中心点定位越准确.因为数据集规模较小,模型在训练

时随机对输入图像进行反转,同时叠加一个服从正态分布的随机白噪声,用以提高模型的鲁棒性.学习率设为0.000 25,训练的 epoch 设为 120,为了防止过拟合,笔者在验证集 Loss 不再下降时停止训练.



图6 数据集采集装置图

表1 训练所用机器配置表

类型	型号	参数
系统	Ubuntu	16.04.2 LTS
CPU	Intel(R) Core(TM) i9-9900	8核
GPU	Nvidia GeForce GTX 1080Ti	11 GB
内存	DDR4	32 GB

### 4.3 实验结果

为了验证本文对 CenterNet 算法提出改进方法的有效性,笔者使用关键点定位精度指标 $AP^{kp}$ ,做了模型自身的纵向对比实验.将关键点位置约束设为改进1,关键点回归自适应补偿设为改进2,实验结果如表2所示,

两项改进均显著提高了 $AP^{kp}$ ,证明了本文方法的有效性.

为了验证本文方法定位托盘中心点的有效性,笔者使用评价指标 $ALP$ ,与其他目标检测算法进行了横向对比实验.对于其他目标检测算法,笔者使用托盘目标框中心点作为托盘中心点的定位结果.实验结果如表3所示.可以看到,本文方法对比其他目标检测算法,中心点定位精度均有显著提高.

表2 纵向对比实验表

算法优化	$AP^{kp}$	$AP_{50}^{kp}$	$AP_{75}^{kp}$
CenterNet	0.352	0.768	0.285
CenterNet+改进1	0.704	0.913	0.745
CenterNet+改进1+改进2	0.728	0.933	0.768

表3 横向对比实验表

方法	$ALP$
FCOS <sup>[19]</sup>	0.916
CenterNet	0.919
YOLOv5 <sup>[20]</sup>	0.920
本文方法	0.946

为验证本文方法估计托盘水平方向的有效性,笔者分别选取托盘左偏、右偏与正对3种具有代表性的情况进行实际测试.结果如图7所示,图7(a)中,关键点 $p_1$ 和 $p_2$ 的y坐标之差 $\Delta y = -75$ ,托盘水平方向左偏;图7(b)中, $\Delta y = 1$ ,托盘水平方向正对;图7(c)中, $\Delta y = 59$ ,托盘水平方向右偏.通过 $\Delta y$ 可有效对托盘水平方向进行估计.



图7 托盘方向估计图

## 5 结论与展望

本文提出了一种基于关键点检测的托盘定位方法.首先将 CenterNet 人体姿态估计引入到托盘定位任务中,完成对托盘外边缘4个关键点的检测.然后改进关键点分组约束,并提出关键点回归自适应补偿,显著提高了托盘关键点定位精度.在关键点检测的基础上,提出了托盘中心点计算与托盘水平方向估计方法,提高了托盘中心点的定位精度,解决了目前目标检测算法无法有效估计托盘水平方向的问题.与其他托盘检测

方法相比,本文提出的方法表现优异,下一步计划将本文方法应用于自动叉车系统中.

### 参考文献

- 赵俊宏, 吕恩利, 陆华忠, 等. 基于激光雷达的托盘位姿识别算法及验证. 仪器仪表学报, 2017, 38(10): 2469-2476. [doi: 10.3969/j.issn.0254-3087.2017.10.015]
- Cui GZ, Lu LS, He ZD, et al. A robust autonomous mobile forklift pallet recognition. Proceedings of the 2nd



- International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2010). Wuhan: IEEE, 2010. 286–290.
- 3 Seelinger M, Yoder JD. Automatic pallet engagement by a vision guided forklift. Proceedings of the 2005 IEEE International Conference on Robotics and Automation. Barcelona: IEEE, 2005. 4068–4073. [doi: [10.1109/ROBOT.2005.1570744](https://doi.org/10.1109/ROBOT.2005.1570744)]
- 4 武文汉. 基于视觉的仓储托盘检测方法研究 [硕士学位论文]. 上海: 上海交通大学, 2018. [doi: [10.27307/d.cnki.gsjtu.2018.002684](https://doi.org/10.27307/d.cnki.gsjtu.2018.002684)]
- 5 Mohamed IS, Capitanni A, Mastrogianni F, *et al.* Detection, localisation and tracking of pallets using machine learning techniques and 2D range data. Neural Computing and Applications, 2020, 32(13): 8811–8828. [doi: [10.1007/s00521-019-04352-0](https://doi.org/10.1007/s00521-019-04352-0)]
- 6 Ren SQ, He KM, Girshick R, *et al.* Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137–1149. [doi: [10.1109/TPAMI.2016.2577031](https://doi.org/10.1109/TPAMI.2016.2577031)]
- 7 李天剑, 黄斌, 刘江玉, 等. 卷积神经网络物体检测算法在物流仓库中的应用. 计算机工程, 2018, 44(6): 176–181. [doi: [10.3969/j.issn.1000-3428.2018.06.031](https://doi.org/10.3969/j.issn.1000-3428.2018.06.031)]
- 8 Liu W, Anguelov D, Erhan D, *et al.* SSD: Single shot multibox detector. Proceedings of the 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 21–37.
- 9 叶伟. 基于视觉的无人叉车导航和托盘定位方法研究与实现 [硕士学位论文]. 合肥: 中国科学技术大学, 2021. [doi: [10.27517/d.cnki.gzkju.2021.001422](https://doi.org/10.27517/d.cnki.gzkju.2021.001422)]
- 10 Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: Optimal speed and accuracy of object detection. arXiv: 2004.10934, 2020.
- 11 Rother C, Kolmogorov V, Blake A. “GrabCut”: Interactive foreground extraction using iterated graph cuts. ACM Transactions on Graphics, 2004, 23(3): 309–314. [doi: [10.1145/1015706.1015720](https://doi.org/10.1145/1015706.1015720)]
- 12 Zhou XY, Wang DQ, Krähenbühl P. Objects as points. arXiv: 1904.07850, 2019.
- 13 Law H, Deng J. CornerNet: Detecting objects as paired keypoints. International Journal of Computer Vision, 2020, 128(3): 642–656. [doi: [10.1007/s11263-019-01204-1](https://doi.org/10.1007/s11263-019-01204-1)]
- 14 Zhou XY, Zhuo JC, Krähenbühl P. Bottom-up object detection by grouping extreme and center points. Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 850–859. [doi: [10.1109/CVPR.2019.00094](https://doi.org/10.1109/CVPR.2019.00094)]
- 15 Newell A, Yang KY, Deng J. Stacked hourglass networks for human pose estimation. Proceedings of 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 483–499.
- 16 He KM, Zhang XY, Ren SQ, *et al.* Deep residual learning for image recognition. Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 770–778. [doi: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90)]
- 17 Yu F, Wang DQ, Shelhamer E, *et al.* Deep layer aggregation. Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 2403–2412. [doi: [10.1109/CVPR.2018.00255](https://doi.org/10.1109/CVPR.2018.00255)]
- 18 Lin TY, Goyal P, Girshick R, *et al.* Focal loss for dense object detection. Proceedings of 2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 2999–3007. [doi: [10.1109/ICCV.2017.324](https://doi.org/10.1109/ICCV.2017.324)]
- 19 Tian Z, Shen CH, Chen H, *et al.* FCOS: Fully convolutional one-stage object detection. Proceedings of 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 9626–9635. [doi: [10.1109/ICCV.2019.00972](https://doi.org/10.1109/ICCV.2019.00972)]
- 20 Jocher G, Stoken A, Borovec J, *et al.* Ultralytics/YOLOv5: V3.0. [https://zenodo.org/record/3983579#\\_YrPiutBByUk](https://zenodo.org/record/3983579#_YrPiutBByUk). (2020-08-13)[2022-01-14]. [doi: [10.5281/zenodo.3983579](https://doi.org/10.5281/zenodo.3983579)]

(校对责编: 孙君艳)