arXiv:2303.03879v1 [cs.CV] 7 Mar 2023

SpinDOE：一种用于乒乓球机器人的球旋转估计方法

Thomas Gossard[1], Jonas Tebbe[1],Andreas Ziegler[1],Andreas Zell[1]

摘要——旋转在乒乓球运动中发挥着重要作用，使击球轨迹更难以读取和预测。然而，由于球的高速运动以及旋转数值的量级，旋转很难测量。现有方法要么需要极高帧率的相机，要么因使用球的标志而不可靠，因为标志并不总是可见。因此，许多乒乓球机器人忽略了旋转，这严重限制了它们的能力。本文提出了一种易于实现且可靠的旋转估计方法。我们开发了一种点状球方向估计（DOE）方法，随后可利用该方法估计旋转。首先使用卷积神经网络在图像中定位圆点，然后通过几何哈希识别这些圆点。最后，根据估计出的姿态回归得到旋转。使用我们的算法，球的姿态估计平均误差为2.4°，旋转估计的相对误差低于 1%。使用 350 fps 的相机可以实时测量最高达 175 转/秒的旋转。借助我们的方法，我们生成了包含位置和旋转信息的乒乓球轨迹数据集，该数据集可在我们的项目页面上获取。

项目页面：https://cogsys-tuebingen.github.io/spindoe/

I. 引言

在乒乓球运动中，旋转估计对于正确回球并赢得比赛至关重要。旋转确实会使球的轨迹变得难以预测。带有上旋的球在反弹后会加速，而施加侧旋时，球的空中轨迹会向侧方弯曲。因此，为了开发乒乓球机器人，旋转估计不仅在比赛过程中至关重要，对于构建准确的乒乓球动力学模型（空气动力学、球台反弹、球拍反弹）也同样重要。

人类球员可以利用不同的线索来估计球的旋转。一些球员能够通过球上标志产生的运动模糊来判断旋转情况 [1]。然而，大多数球员会根据对手的击球动作以及所使用胶皮的先验知识（粘性胶皮、防弧圈胶皮、颗粒胶皮），来估计球上施加了多大的旋转。此外，也可以根据球的空中轨迹和反弹情况估计旋转，不过这种方法留给球员的反应时间更少。

由于旋转估计较为困难，许多乒乓球机器人选择忽略球旋转 [2] [3] [4]，或对其进行“隐式”估计 [5]。只有Tebbe等人 [6] 明确估计球的旋转，并在让机器人进行击球时将其纳入考虑。

目前已有多种旋转估计方法。模仿人类技能的旋转估计主要有三种途径：

[1]作者来自蒂宾根大学信息学系认知系统组。通讯作者 thomas.gossard@uni-tuebingen.de本研究由Sony AI资助。

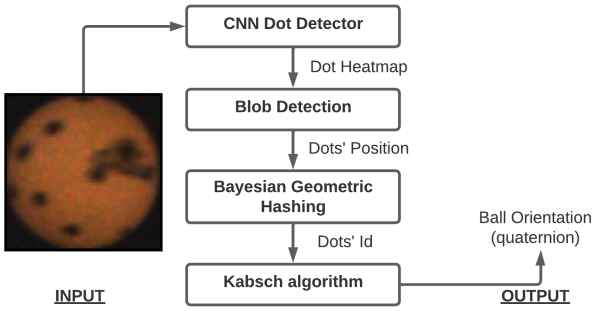


图1：方向估计流程（DOE）

观察击球动作（球拍／身体姿态）、分析球的轨迹，以及直接观察球。

可以根据身体姿态估计 [7][8]，、球拍惯性测量单元测量 [9] [10] 或球拍姿态估计 [11]对击球动作进行分类。这些方法的问题在于，它们只能给出所施加旋转的类型（上旋、下旋、左侧旋或右侧旋、无旋转）。球的确切旋转（旋转向量ω）仍然未知。为了更准确地进行旋转估计，大多数研究都集中于从球的轨迹或球的图像序列中提取旋转。此外，还有一种方法可以从球的标志产生的运动模糊中提取旋转 [12]。然而，这种方法要求模糊基本上由旋转产生，而不是由速度产生。对于乒乓球而言，情况并非如此。事实上，在使用发球机发球时，我们注意到，在旋转值相近的情况下，提高球的速度也会产生运动模糊。

由于使用摄像机直接测量旋转需要较高的帧率，许多研究人员致力于根据马格努斯效应引起的轨迹偏差来估计旋转 [13] [6] [14]。然而，他们很少能够获得旋转真值来计算马格努斯效应系数。此外，旋转估计高度依赖于所记录球的位置的准确性。由于马格努斯效应引起的偏差对于低幅度旋转而言很小，因此这种旋转估计方法在旋转值较低时会对位置测量噪声和偏差高度敏感。

基于观察球的方法能够达到最高的旋转精度。它们可分为两个子类别：基于标志的方法 [15] [6] [16]或基于图案的方法 [17] [18] [19] [20]。基于标志的方法很有价值，因为它们可用于正式乒乓球比赛。这类方法利用标志获取球的方向，然后拟合旋转。

然而，由于标志并非总是可见，而且往往具有一定的对称性，旋转估计常常无法进行或非常困难（例如标志只出现在球的可见部分边缘，或标志靠近旋转轴时）。使用两台摄像机从两个不同视角观察球 [15], 可以弥补这一问题，但也会增加感知系统的复杂性和成本。基于图案的方法因可靠性更高，更适合用于研究。有些方法依赖配准 [17] [18] [20]。使用配准的问题在于，必须有足够高分辨率的图像，才能生成用于唯一识别关键点的描述子。

本文提出了一种新的点状球方向估计方法（DOE），并利用 DOE 估计乒乓球的旋转。因此，我们将该方法命名为 SpinDOE。

SpinDOE 的主要**贡献**包括：

- 使用标准工业相机进行旋转估计。
- 对于最高达 175 转/秒的旋转，具有很高的可靠性和准确性。
- 通过提供 3D打印模板模型和源代码，使该方法易于使用任意类型的球进行复现。
- 证明球在空中飞行时的恒定旋转假设是正确的。
- 生成首个包含精确位置和旋转信息的乒乓球轨迹数据集。

本文接下来将介绍 SpinDOE 的工作原理。在第 II 节中，我们首先介绍方向估计流程。然后在第III 节中，我们展示如何根据一系列姿态回归旋转。在第 IV 节中，我们展示实验结果，例如对旋转阻尼系数的估计。最后一节 V将介绍我们如何创建乒乓球轨迹数据集。

II. 方向估计

我们希望在不妨碍球员的情况下，准确估计每次对打中球的旋转。这意味着相机必须架设在远离球桌的位置，并且需要拥有较大的视野。受这些条件限制，拍摄到的球图像分辨率较低。此外，尽管曝光时间已缩短至最短，高速击球时的球仍会出现运动模糊。因此，无法使用基于配准的方法，也无法跟踪多个关键点。由于我们始终希望仅用一台相机测量球的旋转，因此基于标志的方法也不可行。于是，我们决定使用点图案来估计球的姿态。点图案的优点是不会干扰球体检测流程。我们可以根据圆点的参考三维构型和测得的圆点三维位置，求出它们之间的旋转。该方法的难点在于如何唯一识别这些圆点。实际上，需要识别观测到的圆点，才能将其与对应的参考值关联起来。我们通过几何哈希来实现这一点 [22]。我们没有使用

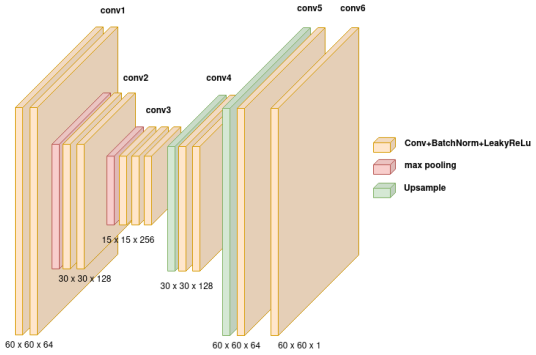


图 2：点检测卷积神经网络架构

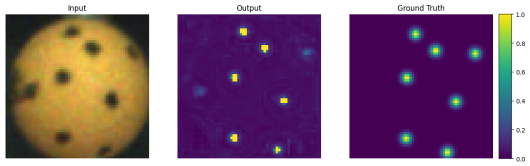


图 3：从左到右：输入图像、输出热力图、真实值热力图

图像特征来识别它们（如配准方法所做的那样），而是利用它们的空间构型。

DOE 的工作流程如下：卷积神经网络首先输入球的图像，并输出圆点可能位置的热力图。卷积神经网络的圆点检测基于 CenterNet [23]。随后通过斑点检测从热力图中提取圆点位置。为了将检测到的圆点与参考图案中的圆点匹配，采用了几何哈希。最后，我们可以使用Kabsch算法求得球的方向（Kabsch算法根据均方根误差，计算两组向量之间的最优旋转）。

A. 点检测

我们首先测试了传统的计算机视觉方法，例如背景减除、掩模处理和斑点检测，以估计图像中圆点的位置。经过标定后，这些方法取得了相对较好的结果，但对图像变化（亮度、对比度、颜色）也非常敏感。因此，我们改用基于 CenterNet 的卷积神经网络[23]。具体架构见图2。这使点检测更加稳健。使用卷积神经网络也使我们能够使用带有标志的球。事实上，卷积神经网络会学会忽略标志（如图8最左侧的图像所示），但在标志仅部分显示于边缘等复杂情况下除外。因此，我们的方法适用于任何类型的球，不过针对特定标志，卷积神经网络可能需要进行微调。图3展示了卷积神经网络圆点检测器的输入、输出和真实值示例。

B. 数据集

为生成数据集，使用无刷直流电机使带圆点的球旋转（图。4）。电机由ODrive控制器控制。之所以选择 ODrive，是因为其具备磁场定向控制（FOC）功能。通过该控制器，可以准确测量旋转的范数（测量噪声为 0.1 转/秒），并将其作为真实值。

表 I：现有　　基于标志或图案的旋转估计方法汇总（论文按发表时间顺序排列）　　r)

算法		方法	最大自旋速度（报告值）[转/秒]	相机帧率	相对误差 [%]
Tamaki 等人	[17]	图案（配准）	90	500	NA
Theobalt 等人	[21]	图案（彩色标记）	27	80	NA
Furuno等人	[19]	图案（彩色线条）	23	1200	NA
Boracchi等人	[12]	Blur	28	NA	3
Sz`ep	[20]	图案（轨迹角点）	63	1000	12.5
Tamaki 等	[18]	图案	67	600	NA
Glover等人	[16]	标志*	50	200	NA
Zhang等人	[15]	Logo	60	NA	0.07
Tebbe等人	[6]	Logo	75	380	NA
我们的方法		图案	175	350	1

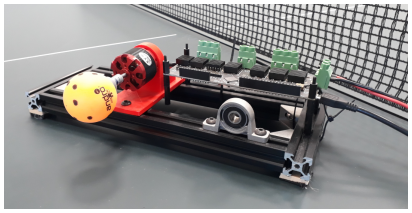


图 4：用于生成训练数据的球旋转器

为准确获得旋转轴，安装球体时使旋转轴与所绘制的一个圆点重合。在此设置下，我们获得了真实值（旋转向量），并生成了旋转球图像数据集。这些记录被用作旋转估计方法的基准测试工具，以及训练点检测网络的数据集。为获得图像中圆点的真实位置，将圆点的参考三维位置根据球体计算得到的姿态进行旋转。球体姿态通过将初始人工测量姿态与对应的旋转进行传播而获得。估计的圆点位置与实际圆点位置之间通常存在一些小偏移。为进行补偿，将估计的圆点位置校正到最近的局部灰度极小值处，即黑色圆点处。此外，每 Δt 次测量都会重置初始校正，以避免旋转和初始姿态产生的误差传播到估计的球体姿态中。我们为具有两种不同标志的球生成了一个包含 80,000 个样本的数据集，旋转范围为 0 至 150 转/秒。为提高点检测器的鲁棒性，通过添加运动模糊以及改变亮度、对比度、饱和度和色调对数据进行了增强。

C. 贝叶斯几何哈希

几何哈希是一种物体识别方法，它利用关键点的空间排列来识别物体。该方法使我们能够识别圆点（获取其索引），尽管这些圆点完全相同且分辨率较低。

几何哈希可分为两个步骤。首先，根据我们想要识别的参考物体生成一个查找表，即哈希表。然后使用该查找表识别物体。

生成哈希表： 将物体的关键点 $\boldsymbol{D}=\{\boldsymbol{d_1},\boldsymbol{d_2},...\}$ 转换到一个称为哈希空间、具有旋转、尺度和平移不变性的空间中。具体做法是使用两个关键点（在二维情况下）作为基，描述其他特征的相对位置。对每个关键点以及每种可能的基组合，都执行这一过程。最终，我们得到一个表，其中每个条目都包含用于构成基的关键点，以及一个向量 $\boldsymbol{h}$；该向量表示参考模型中的另一个关键点位置，并依据指定的基转换到哈希空间中。

识别： 首先从输入图像中提取关键点 $\boldsymbol{D}$。从这些关键点中，任意选择 2 个作为基，将其他关键点转换到哈希空间中。每个转换后的关键点 $\boldsymbol{\phi}$ 都会在哈希空间中为附近的哈希值 $\boldsymbol{h}$ 以及特定基 $\boldsymbol{basis_h}$投票。为此，需要将哈希值分箱，转换后的关键点则为与其处于同一箱中的哈希值投票。得票数高于某一阈值的基将被预选，并计算其重投影均方根误差。选择均方根误差最小的基作为识别出的模型。

在我们的案例中，我们处理的是三维点。由于我们知道圆点位于球的表面（$z=\sqrt{r-x^2-y^2}$），因此可以根据它们的三维图像位置计算出圆点的三维坐标。球心和两个圆点被用作描述其他圆点所需的基。

传统的几何哈希方法使用分箱进行基投票。不过，也存在一种贝叶斯变体[22]。我们不再预先选取得票最多的基，而是选择似然最高的基。这使几何哈希更加稳健，因为连续概率分布保证了“投票”的连续性，这与分箱不同。

在算法2中，$p_\phi(\boldsymbol{h})$ 表示特征 $\boldsymbol{\phi}$（通过函数 $f(\boldsymbol{x})=\boldsymbol{B}^{-1}\cdot\boldsymbol{x}$将圆点变换到哈希空间中）对应于哈希值$\boldsymbol{h}$（将参考圆点变换到哈希空间中）。我们希望该似然能够表示圆点在哈希空间中的位置不确定性。为此，我们使用变量变换 f，将球面上的圆点位置不确定性转换为哈希空间中的等效不确定性，如公式3所示。不确定性可能由多种因素造成：圆点绘制不准确、卷积神经网络圆点检测器不够精确以及运动模糊。我们使用Kent分布对圆点位置不确定性进行建模

算法 1 几何哈希：生成哈希表

要求： 参考点位置： $\boldsymbol{D} = [\boldsymbol{d}_1, \boldsymbol{d}_2, ...] \in \mathbb{R}^{3\times n}$
  $\boldsymbol{hash_table} = []$
  $\boldsymbol{basis_used} = []$
  对于 d，在 D **中执行**
    对于 d'，在 $D \setminus d$ **中执行**
      $\boldsymbol{basis} = [\boldsymbol{d}, \boldsymbol{d}', \boldsymbol{d} \times \boldsymbol{d}']$
      对于 d''，在 $D \setminus \{d, d'\}$ **中执行**
        $\boldsymbol{h_i} = \boldsymbol{basis}^{-1} \cdot \boldsymbol{d''}$
        将 $\boldsymbol{h_i}$ 追加到 $\boldsymbol{hash_table}$
        将 $\boldsymbol{basis}$追加到 $\boldsymbol{bases_used}$
      结束循环
    结束循环
  结束循环

算法 2 贝叶斯几何哈希：识别

需要： 输入点位置： $\boldsymbol{D} = [\boldsymbol{d}_1, \boldsymbol{d}_2, ...] \in \mathbb{R}^{3\times n}$
  $\boldsymbol{basis} = [\boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{d}_1 \times \boldsymbol{d}_2]$
  $\boldsymbol{D'} \leftarrow D \setminus \{\boldsymbol{d_1}, \boldsymbol{d_2}\}$
  $\boldsymbol{\Phi} \leftarrow \boldsymbol{basis}^{-1} \cdot \boldsymbol{D'}$ ▷ 将圆点转换到哈希空间
  $\boldsymbol{poss_bases} = []$ 可能基列表
  对 $\phi \in \boldsymbol{\Phi}$ **执行**
    $\boldsymbol{H} \leftarrow NearestHashValues(\boldsymbol{\phi})$
    对 h **属于** $\boldsymbol{H}$ **执行**
      如果 $\boldsymbol{basis_h}$ 不属于 $\boldsymbol{poss\ bases}$**则**
        将 $\boldsymbol{basis_h}$ 追加到 $\boldsymbol{poss_bases}$
      结束 if
      $\boldsymbol{score_{basis_h}} += p_{\boldsymbol{\phi}}(\boldsymbol{h})$
    结束 for
  结束 for
  $\boldsymbol{selected\ bases}$ 是 $\boldsymbol{basis}$，其中 $score \geq threshold$
  对于 $\boldsymbol{basis}$ 中的 $\boldsymbol{selected\ bases}$**执行**
    $\boldsymbol{rot} \leftarrow Kabsch(\boldsymbol{basis}, \boldsymbol{D})$
    $error \leftarrow reprojection\ error(rot, \boldsymbol{D})$
  结束循环
  返回 $\boldsymbol{rot}$ 误差最小的那个

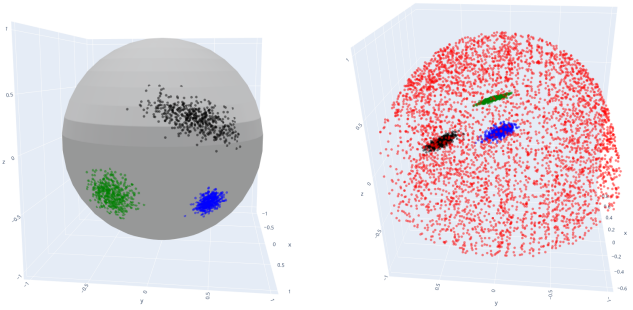
在球面上 $p_{\boldsymbol{d}}(\boldsymbol{x})$。Kent分布可以看作球面上的正态分布，其描述如下：

$$k_{\boldsymbol{d}}(\boldsymbol{x}) = \frac{1}{\mathrm{c}(\kappa,\beta)} \exp\left\{\kappa\gamma_1^T \cdot \boldsymbol{x} + \beta\left[\left(\gamma_2^T \cdot \boldsymbol{x}\right)^2 - \left(\gamma_3^T \cdot \boldsymbol{x}\right)^2\right]\right\} \tag{1}$$

其中：

$$c(\kappa,\beta) = 2\pi \sum_{j=0}^{\infty} \frac{\Gamma\left(j+\frac{1}{2}\right)}{\Gamma(j+1)} \beta^{2j} \left(\frac{1}{2}\kappa\right)^{-2j-\frac{1}{2}} I_{2j+\frac{1}{2}}(\kappa) \tag{2}$$

其中，$I_v(\kappa)$ 是修正贝塞尔函数，$\Gamma(\cdot)$ 是伽马函数，$\boldsymbol{\gamma_1}$，$\boldsymbol{\gamma_2}$，$\boldsymbol{\gamma_3}$ 分别是表示概率密度函数均值轴、主轴和次轴的正交单位向量（它们根据 $\boldsymbol{d}$ 计算得到），κ 决定概率密度函数的集中度，β 决定概率密度函数的偏心率。这些超参数根据经验设置为$\kappa = 500$ 和 $\beta = 0$，使概率密度函数与该点的



(a) 球面空间　　(b) 哈希空间

图 5：**Kent分布**：在球体上绘制了不同位置对应的样本（黑色：$\kappa = 100, \beta = 40$，蓝色：$\kappa = 300, \beta = 0$，绿色：$\kappa = 100, \beta = 0$）。这些样本通过另外两个关键点变换到哈希空间。红色点是为哈希表生成的所有哈希值。

大小和形状相对应。具有不同参数的 Kent分布示例如图5a所示。

为得到 $p_{\boldsymbol{\phi}}(\boldsymbol{h})$，我们将哈希空间中的 Kent 分布变换如下：

$$\begin{aligned} p_{\boldsymbol{\phi}}(\boldsymbol{h}) &= p_{\boldsymbol{d}}(f^{-1}(\boldsymbol{h}))|det\frac{\partial f^{-1}(\boldsymbol{h})}{\partial \boldsymbol{h}}| \\ &= p_{\boldsymbol{d}}(\boldsymbol{B}\cdot\boldsymbol{h})|det(\boldsymbol{B})| \end{aligned} \tag{3}$$

反函数的雅可比矩阵行列式的范数确保在变量变换过程中概率分布的体积保持不变。

然而，我们需要添加一个“投影”似然 $n(\boldsymbol{x})$，以便点位置不确定性概率密度函数 $p_{\boldsymbol{d}}$ 能够在三维哈希空间中发挥作用（Kent分布作用于二维球面流形）。投影 $n(\boldsymbol{x})$ 设定为点位于球体表面（范数为 1）的似然。

$$p_{\boldsymbol{d}}(\boldsymbol{x}) = n(\boldsymbol{x})k_{\boldsymbol{d}}(\boldsymbol{x}) \tag{4}$$

其中：

$$n(\boldsymbol{x}) = \frac{1}{\alpha\sqrt{2\pi}} \exp\left(-\frac{1}{2}\frac{(||\boldsymbol{x}|| - 1)^2}{\alpha^2}\right) \tag{5}$$

且 α 是我们“投影”分量的标准差。$\alpha = 0.03$ 该参数的选择旨在最大化几何哈希的识别率，其示意图见图6。我们确实使用蒙特卡洛测试研究了贝叶斯几何哈希的敏感性。下一节将更详细地介绍所用的方法。

D. 图案生成

我们使用 3D 打印的模板在球上绘制点图案（图7）。这使我们能够准确、可靠地将圆点定位在球的表面上。这种方法还使我们能够轻松、低成本地为机器人乒乓球装置制作多个球。

圆点的数量被任意设定为 20 个。这使得从任意角度都能看到的点平均超过 3 个，而这

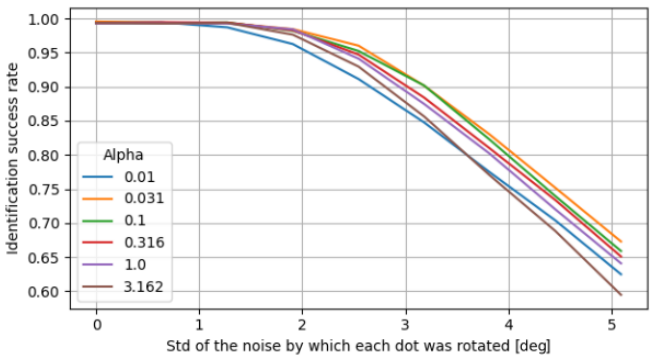


图 6：几何哈希对圆点位置不准确的敏感性

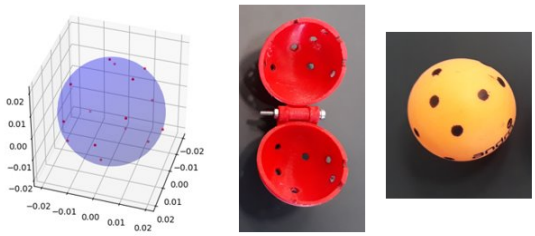


图 7：从左到右：生成的图案、3D打印模具、带有点图案的球

对于使用几何哈希是必要的。另一方面，圆点的位置并非任意设定。通过几何哈希，可以根据特征的空间配置识别物体。这意味着并非所有图案都具有相同的效果，其中一些图案对噪声具有更好的鲁棒性。

使用蒙特卡洛测试的成功率评估图案的鲁棒性。单次测试流程如下：首先，对原始点配置应用随机旋转，并仅选取可见的圆点（$d_z > 0$ 假设 $\vec{z}$ 朝向我们）。此举旨在测试不同的观察角度。随后，通过施加小幅随机旋转，分别为每个可见圆点添加噪声。最后，将生成的圆点集输入几何哈希。使用含有 $\sigma = 3°$噪声的 10 000 个样本对图案进行评估。我们认为，这种噪声最能代表圆点位置误差。

我们选择基于梯度的优化方法来寻找最佳图案。我们通过最大化哈希空间中特征的平均最近邻距离，对位于单位球面上的圆点的球面坐标进行优化$(\theta_i,\ \Phi_i)$。这样可以使哈希空间中的特征更易于区分，从而提高其对噪声的鲁棒性。因此，我们生成了一种点图案，在噪声标准差为 3 °时，其识别成功率达到 98%，如图6所示。

III. 旋转估计

A. 旋转回归

尽管点检测和贝叶斯几何哈希具有较高的识别成功率，但误识别并非不可能。

我们将 RANSAC 与旋转回归结合，使我们的算法能够稳健地应对可能存在的离群点。对于旋转回归，我们使用 QuateRA [24],基于四元数的旋转回归算法。根据测量结果，使用奇异值分解计算旋转平面。这将旋转轴确定为垂直于旋转平面的向量。随后，将每个四元数投影到旋转平面上。在此过程中，使用 RANSAC 选择用于计算旋转平面的有效测量值。然后，通过对连续旋转角进行线性回归来估计旋转。这种方法与 [6]非常相似。主要区别在于，QuateRA 使用奇异值分解以代数方式计算旋转平面，这也是最小二乘解。在 [6], 中，旋转平面是与三个标志位置相交的平面。选择这三个标志位置，以使其他标志位置的投影误差最小。与[6]相比，这使我们的旋转回归速度更快且更加准确。

IV. 实验

A. 设置

使用 Grasshopper3 GS3-U3-23S6C 相机以 350 fps 的帧率和 1900x400 的分辨率采集球图像。曝光时间设置为 250 μs。使用 [25], 中描述的方法提取球的感兴趣区域，随后获得分辨率为 (60x60) 的图像。我们的旋转估计方法 SpinDOE 在一台配备 NVIDIA GeForce GTX 1080 Ti GPU 和 Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz 的计算机上运行。

B. 方向估计

我们首先测试了方向估计方法。借助球旋转器，我们生成了一个包含球图像和姿态的基准数据集。我们在图8中给出了 DOE 输出的一个视觉示例。如图所示，卷积神经网络忽略了标志，仅在热力图中为圆点输出较高值。我们还可以注意到，卷积神经网络能够检测到球边缘上的圆点；而使用传统的计算机视觉算法则无法做到这一点。

在图9中，我们绘制了方向估计误差的分布。如预期所料，存在方向估计失败的情况。这主要是由于球上标志的存在，该标志有时会被检测为一个或两个圆点。DOE 对球是否位于图像中心也很敏感。事实上，图像中心被假定为球的中心，并据此计算圆点的三维位置。当只检测到两个或三个圆点时，也会出现一些估计失败的情况。尽管如此，在整个基准数据集上，失败率（误差超过 20 °）为 7%。这使我们的平均误差达到 2.3 ± 2.3 °如果忽略识别失败。

就处理速度而言，DOE处理 10 张图像（旋转回归所需的球图像数量）需要 0.043 ± 0.002 s。

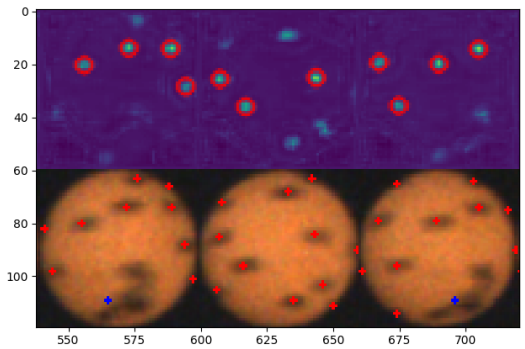


图1 8：转/秒为 150 的球旋转器测试案例 - **第 1 行：**生成的热力图，通过斑点检测定位的圆点以红色圈出；**第 2 行：**输入图像，图中叠加了估计的圆点位置（红色圆点表示估计的圆点位置，蓝色圆点表示估计的标志位置）

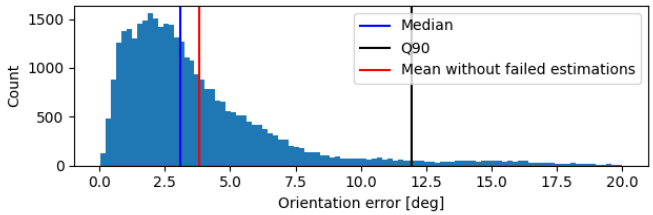


图1 9：DOE 的误差分布（范围限制在0°和 20°之间，以提高可读性）

C. 旋转估计

我们使用球旋转器基准数据集评估了旋转估计方法的准确性。我们限定输入为十张图像，相当于 26 ms 的测量时长。如图10 所示，SpinDOE 有 90% 的情况下相对误差低于 0.20。大多数错误的旋转估计出现在低旋转值（低于 10 转/秒）时，这是因为相对误差增大；以及高旋转值（高于 140 转/秒）时，这是因为旋转回归中的角度展开失败。对于十张图像的输入，RANSAC旋转回归的运行时间为 $0.017 \pm 0.002\ s$。SpinDOE 总共需要 $0.061 \pm 0.003\ s$ 才能输出旋转值。这足以让 SpinDOE 实时运行。

1）恒定旋转假设：机器人学中关于乒乓球最常见的假设之一是球的旋转

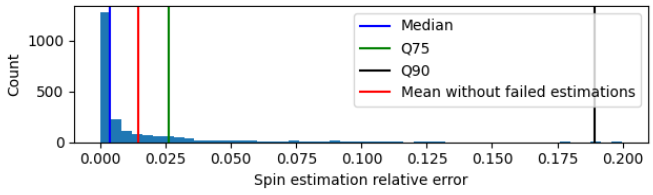


图1 10：SpinDOE 的相对误差分布（范围限制在 0 和 0.20 之间，以提高可读性）

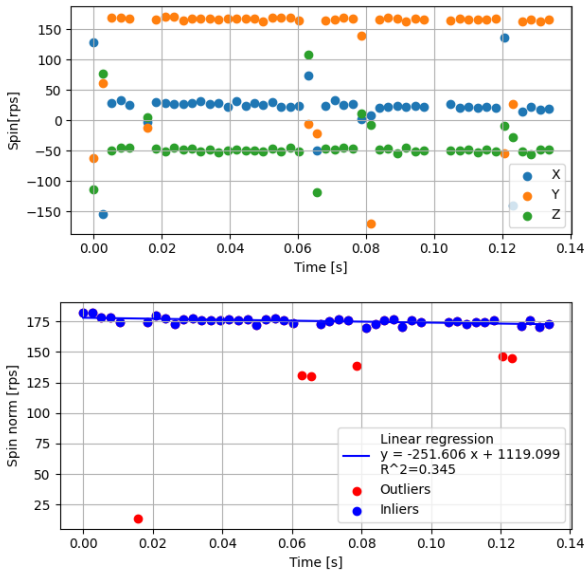


图1 11：根据连续两次球姿态之间的差值计算出的旋转

是恒定的。然而，据我们所知，这一假设从未得到过实证验证。从理论上讲，空气会对所有旋转球体施加黏性力矩[26]，使其旋转速度减慢。球的旋转动力学可以用以下方程描述：

$$I\frac{\omega(t)}{dt} = \frac{2}{4}mr^3\frac{\omega(t)}{dt} = T_{viscous} = -8\pi\nu r^3\boldsymbol{\omega}(t) \quad (6)$$

其中，I 是球的转动惯量，ν 是空气黏度，m 是球的质量，r 是球的半径。

求解该常微分方程可得到旋转演变：

$$\begin{aligned}\omega(t) &= \omega(t_0)\exp\left(-\frac{12\pi\nu r}{m}t\right) \\ &\approx \omega(t_0) - \omega(t_0)\frac{12\pi\nu r}{m}t\end{aligned} \quad (7)$$

阻尼系数为：$-\frac{12\pi\nu r}{m} \approx -0.005$（使用 $\nu = 1.81\times10^{-5}$ kg/(m·s)、$m = 2.7g$ 和 $r = 20$mm 计算得到的理论值）。

在图11中，我们展示了使用 Butterfly Amicus 发球机以最大旋转速度和速度设置发出的球的旋转情况。从范数曲线可以看出，旋转确实在减小，而线性回归表明，旋转衰减的理论值被严重低估了。然而，相对于飞行时间而言，旋转的减小非常轻微，因此恒定旋转假设仍然成立。

我们尝试评估多次发球的旋转衰减系数，并得到均值 0.091 ± 0.03。这表明理论值明显被低估了。

V. 数据集

使用 SpinDOE，我们生成了一个乒乓球轨迹数据集。球的位置以 145 赫兹的频率记录，

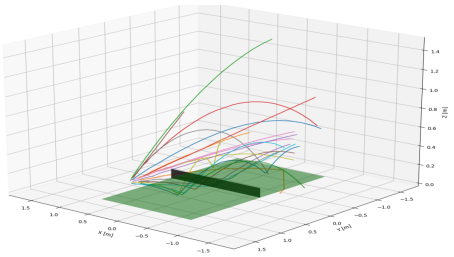


图12：记录数据集中的部分轨迹

旋转仅记录一次，因此只在第一次弹跳前有效。据我们所知，目前没有其他类似的数据集公开提供。球的位置使用 [25] 中所述的方法进行捕捉。使用两台摄像机对球的位置进行三角测量。通过标准计算机视觉算法（背景减除、掩膜生成和斑点检测）提取球在图像中的位置。

该数据集包含200条轨迹，其中一部分如图12所示。观测到的最大速度和旋转分别为11 m/s和150转/秒。这些球由Amicus发球机发出，其发球设置采用均匀采样。

该数据集可用于多种场景：例如，为轨迹预测算法提供基准，检验根据球轨迹估计旋转的算法是否确实有效，或者构建贝叶斯乒乓球模拟器，以减小强化学习中的仿真到现实差距。

VI. 结论

尽管我们的方法无法用于正式乒乓球比赛中的旋转测量，但它对研究非常有用。为此，我们提供了一个包含球旋转信息的球轨迹数据集。与其他方法相比，SpinDOE不要求高分辨率或高帧率。它还能够稳健地应对运动模糊，并且不会像基于标志的方法那样受到隐藏标记的影响。最重要的是，它具有很高的准确性，例如，可以观察到黏性力矩使球的旋转逐渐减慢的效果。该方法应用于乒乓球机器人领域，但也可以用于网球、手球或足球等其他球类运动。

参考文献

[1] “(S+) 蒂莫·波尔：《我把印章固定在球上》”，《明镜》，2018年9月。[2] S. Abeyruwan、L. Graesser、D. B. D’Ambrosio、A. Singh、A. Shankar、A. Bewley 和 P. R. Sanketi，《I-Sim2Real：在紧密的人机交互闭环中学习机器人策略》，2022年8月。[3] Y. Zhang、R. Xiong、Y. Zhao 和 J. Chu，乒乓球机器人的自适应轨迹预测方法，2012年10月。[4] D. B¨uchler、S. Guist、R. Calandra、V. Berenz、B. Sch¨olkopf 和 J. Peters，《使用肌肉机器人从零开始学习打乒乓球》，2020年6月。[5] L. Yang、H. Zhang、X. Zhu 和 X. Sheng，《基于时间序列深度强化学习的乒乓球机器人系统球运动控制》，*IEEE Access*，第9卷，第99 816–99 827页，2021年。

[6] J. Tebbe、L. Klamt、Y. Gao 和 A. Zell，《机器人乒乓球中的旋转检测》，*2020*年*IEEE*国际机器人与自动化会议（*ICRA*），第9694–9700页，2020年5月。[7] S. Sato 和 M. Aono，《利用人体姿态估计模型进行乒乓球击球分类》，第3页。[8] K. M. Kulkarni 和 S. Shenoy，《使用二维人体姿态估计进行乒乓球击球识别》，2021年5月。[9] P. Blank、J. Hoßbach、D. Schuldhaus 和 B. M. Eskofier，《乒乓球中的传感器式击球检测与击球类型分类》，载于*2015*年*ACM*可穿戴计算国际研讨会论文集，系列：ISWC ’15。美国纽约州纽约：计算机协会，2015年9月，第93–100页。[10] P. Blank、B. H. Groh 和 B. M. Eskofier，《使用球拍安装式惯性传感器估计乒乓球的球速和旋转》，载于*2017*年*ACM*可穿戴计算国际研讨会论文集，系列：ISWC ’17。美国纽约州纽约：计算机协会，2017年9月，第2–9页。[11] Y. Gao、J. Tebbe 和 A. Zell，《通过视觉和惯性测量单元实现机器人乒乓球中的鲁棒击球识别》，载于人工神经网络与机器学习——*ICANN 2021*，系列：《计算机科学讲义》，I. Farkaˇs、P. Masulli、S. Otte 和 S. Wermter 编。查姆：施普林格国际出版，2021年，第379–390页。[12] 《从运动模糊中单图像三维重建球速和旋转——基于运动模糊的运动恢复实验》，载于第三届计算机视觉理论与应用国际会议论文集。葡萄牙马德拉丰沙尔：SciTePress - 科学与技术出版物，2008年，第22–29页。[13] X. Chen、Y. Tian、Q. Huang、W. Zhang 和 Z. Yu，《面向机器人乒乓球选手的基于动力学模型的球轨迹预测》，*2010*年*IEEE*国际机器人与仿生学会议，*ROBIO 2010*，2010年12月。[14] H. Su、Z. Fang、D. Xu 和 M. Tan，《基于模糊滤波和局部建模的机器人乒乓球选手旋转球轨迹预测》，*IEEE*《仪器与测量汇刊》，第62卷，第11期，第2890–2900页，2013年11月。[15] Y. Zhang、R. Xiong、Y. Zhao 和 J. Wang，《利用乒乓球的天然标记实时估计其旋转》，*IEEE*《仪器与测量汇刊》，第64卷，第1–1页，2015年8月。[16] J. Glover 和 L. P. Kaelbling，《使用四元数Bingham滤波器跟踪乒乓球的旋转》，载于*2014*年*IEEE*国际机器人与自动化会议（*ICRA*）。中国香港：IEEE，2014年5月，第4133–4140页。[17] T. Tamaki、T. Sugino 和 M. Yamamoto，《通过图像配准测量球的旋转》，2004年3月。[18] T. Tamaki、H. Wang、B. Raytchev、K. Kaneda 和 Y. Ushiyama，《使用逆合成图像对齐估计乒乓球的旋转》，载于声学、语音与信号处理，*1988*年。*ICASSP-88*，*1988*年国际会议，2012年3月，第1457–1460页。[19] S. Furuno、K. Kobayashi、T. Okubo 和 Y. Kurihara，《使用带有独特标记的运动球测量旋转速率的研究》，载于*2009 ICCAS-SICE*，2009年8月，第3439–3442页。[20] A. Sz`ep，《在单目视频中测量球的旋转》，2011年2月。[21] C. Theobalt、I. Albrecht、J. Haber、M. Magnor 和 H.-P. Seidel，《投掷棒球——使用多重曝光图像跟踪高速运动》，第8页。[22] H. Wolfson 和 I. Rigoutsos，《几何哈希：综述》，*IEEE*《计算科学与工程》，第4卷，第10–21页，1997年11月。[23] K. Duan、S. Bai、L. Xie、H. Qi、Q. Huang 和 Q. Tian，《CenterNet：用于目标检测的关键点三元组》，2019年4月。[24] M. M. de Almeida、D. Mortari、R. Zanetti 和 M. Akella，《QuateRA：四元数回归算法》，《制导、控制与动力学杂志》，第43卷，第9期，第1600–1616页，2020年9月。[25] J. Tebbe、Y. Gao、M. Sastre-Rienietz 和 A. Zell，《使用工业KUKA机械臂的乒乓球机器人系统》，载于模式识别，系列：《计算机科学讲义》，T. Brox、A. Bruhn 和 M. Fritz 编。查姆：施普林格国际出版，2019年，第33–45页。[26] U. Lei、C. Y. Yang 和 K. C. Wu，《任意旋转下球体的黏性力矩》，应用物理快报，第89卷，第18期，第181908页，2006年10月。