

```
url: "https://seimei.do/zh/blog/taper-measurement-lidar-scan-attempt-that-failed/"
title: "锥度测量：最先尝试的 iPhone LiDAR 扫描方案失败了"
description: "在手机长焦相机测量锥度角之前，我最先尝试的其实是 3D 扫描。iPhone LiDAR / ObjectCapture  
→ 的原型始终没能进入正式采集状态。记录这条走不通的路。"
pubDate: 2026-07-10
tags: ["measurement", "3d-scanning", "failure", "manufacturing"]
series: "constella-measurement"
related: ["smartphone-telephoto-taper-measurement",  
→ "stable-ts-faster-whisper-local-transcription-comparison"]
graphLabel: "锥度测量 LiDAR 扫描失败"
graphWeight: 1
ogImage: "/blog/smartphone-telephoto-taper-measurement/og.jpg"
lang: zh
```

<https://seimei.do/zh/blog/taper-measurement-lidar-scan-attempt-that-failed/>

课题

上一篇文章讲的是用手机长焦相机测量锥度角，其实在那之前还有一段前情。伪平行投影并不是我最先想到的办法。最先动手的是一条看起来更直接的路。扫描零件，对点云拟合圆锥，一步到位：角度和直径一次到手，看起来也不必细抠拍摄条件。iPhone 上正好有 LiDAR。按常理，先走的当然是这条路。

只想看结论的话，请跳转到小结。

起因是看到了 WIDAR 这款消费级 3D 扫描应用（截至写作时已停止服务）。界面、分享、导出这些平台层的东西姑且不论，我判断其核心扫描能力应该只是 Apple 自家 ObjectCapture / ARKit 框架的一层薄封装。既然如此，同等功能自己就能搭出来。于是动手了。

思路

把 ObjectCapture 和 ARKit 的 LiDAR 网格导出组合起来：绕零件走一圈拍摄，重建 3D 网格，再从网格上拟合锥面。ObjectCapture 的卖点正是如此：不需要专用夹具，拿着手机绕对象移动就能得到 3D 数据。流程图几乎是自己画出来的。

实现起来确实容易

「核心只是标准框架的薄封装」这个判断没有错。启动 ObjectCapture 会话、导出 LiDAR 网格的 App，按框架自带的状态机（检测、采集、重建）写下来就成型了。到这里为止毫无波澜。老实说，我以为这事已经稳了。

失败

把 App 对准实际零件之后，它连采集状态都进不去。`startDetecting` 一直返回 `false`，反馈只有 `movingTooFast`（移动过快）和 `environmentLowLight`（光线不足）反复出现。放慢动作、调整光照、调整距离，拍摄张数始终为零。

实现已经完工，却寸步难行。代码里没有可疑之处，原因反而更难锁定。

诊断

深入排查之后发现，这不是靠调条件能翻过去的墙。

分辨率不够。已公开的评测显示，iPhone LiDAR 的实用精度大致在 10 厘米量级及以上。毫米级的锥度细节，根本就在这颗传感器的分辨能力之外。

没有特征点。ObjectCapture 的摄影测量靠在多张图像间匹配表面特征来重建网格。这个零件小、光滑、轴对称，几乎是摄影测量的最坏情况：表面上没有任何可供帧间匹配的线索。我现在的理解是，`movingTooFast` 这个警告和移动速度无关，它是追踪器丢失特征后位姿估计崩溃的外在表现。

换句话说，问题不在实现。传感器分辨率与零件形状这一物理层面，决定了这条路从一开始就走不通。

转向

我也找过退路：给 iPhone 外接一枚廉价的远心转换镜头。市面上没有这种产品。

于是把问题往后退一步看。零件是轴对称的锥面，真正要的不是 3D 点云，而是一个标量：剪影两条边缘线的夹角。既然如此，3D 重建根本就是多余的。降掉一个维度，改从单张静态照片的剪影里读角度。这就引出了上一篇文章中用长焦拉开距离、逼近伪平行投影的方法。

小结

这次失败的教训不是「LiDAR 扫描没玩好」，而是：所选原理的维度相对于想测的量本来就过剩，而且这套原理和零件形状几乎是最坏搭配。实现如预期般容易。但「容不容易实现」和「原理在你的对象上成不成立」是两条完全独立的轴。

手头没有专用仪器时，第一个该问的不是「能不能轻松搭个替代品」，而是「这件事到底需要几维信息」「这套原理在这个形状上成不成立」。要是先回答了这两个问题，这个 App 根本不用写。不过话说回来，正因为写了它，我才能讲清楚下一个方法为什么是对的。作为学费，不算贵。

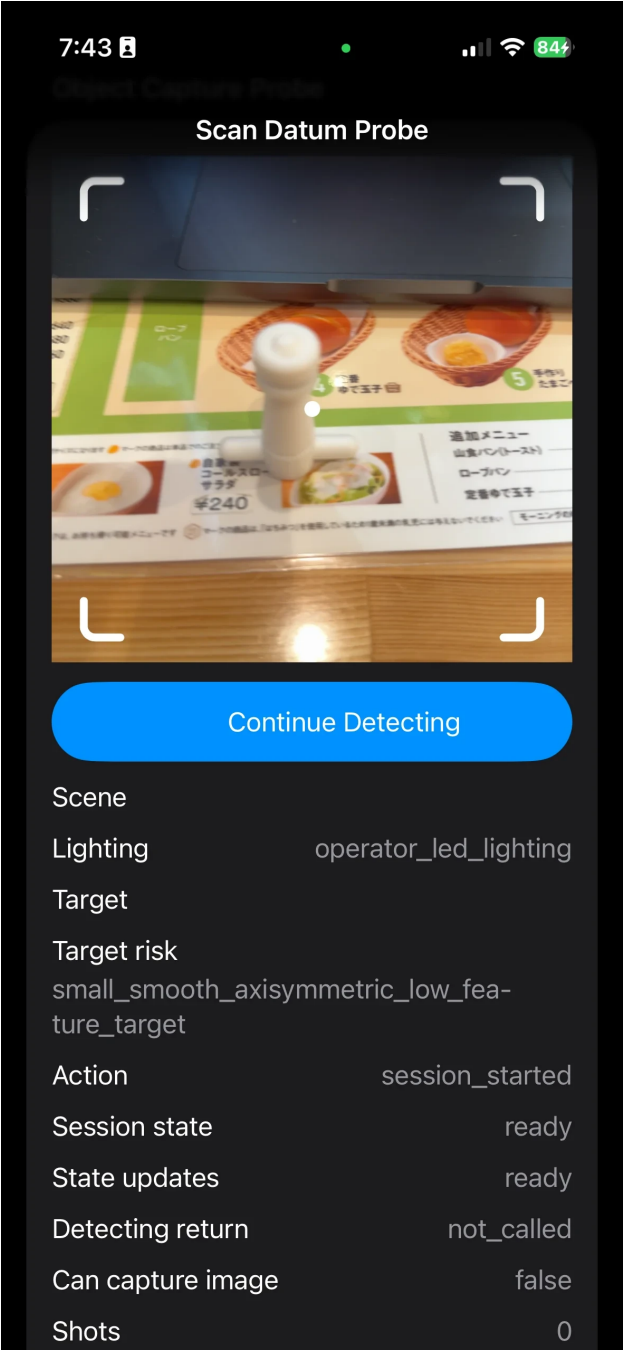


图 1 实机上的 ObjectCapture 调试画面。摄像头画面中是零件 (一个活栓)，底部 HUD 显示 Action: session_started、Detecting return: not_called、Can capture image: false、Shots: 0 (内部标识符已打码)