MediaPipe Hands：设备上实时手部跟踪

Fan Zhang Valentin Bazarevsky Andrey Vakunov

Andrei Tkachenka George Sung Chuo-Ling Chang Matthias Grundmann

Google Research

1600 Amphitheatre Pkwy, Mountain View, CA 94043, USA

{zhafang, valik, vakunov, atkach, gsung, chuoling, grundman}@google.com

# 摘要

我们提出了一种实时设备手部跟踪解决方案，可通过单个 RGB 摄像头预测人类手部骨架，适用于 AR/VR 应用。 我们的管道由两个模型组成： 1）手掌检测器，提供手部的边界框；2）手部地标模型，预测手部骨架。 它是通过 MediaPipe[12] 实现的，这是一个用于构建跨平台 ML 解决方案的框架。 所提出的模型和流水线架构在移动 GPU 上展示了实时推理速度和较高的预测质量。 MediaPipe Hands开源于*<https://mediapipe.dev>*。

# 引言

手部跟踪是在 AR/VR 中提供自然交互和交流方式的重要组成部分，一直是业界活跃的研究课题 [2] [15]。 基于视觉的手部姿势估计已研究多年。 以前的大部分工作都需要专门的硬件，例如深度传感器[13][16][17][3][4]。 其他解决方案不够轻便，无法在普通移动设备上实时运行[5]，因此仅限于配备强大处理器的平台。 在本文中，我们提出了一种新颖的解决方案，它不需要任何额外的硬件，就能在移动设备上实时运行。 我们的主要贡献是:

arXiv:2006.10214v1 [cs.CV] 18 Jun 2020

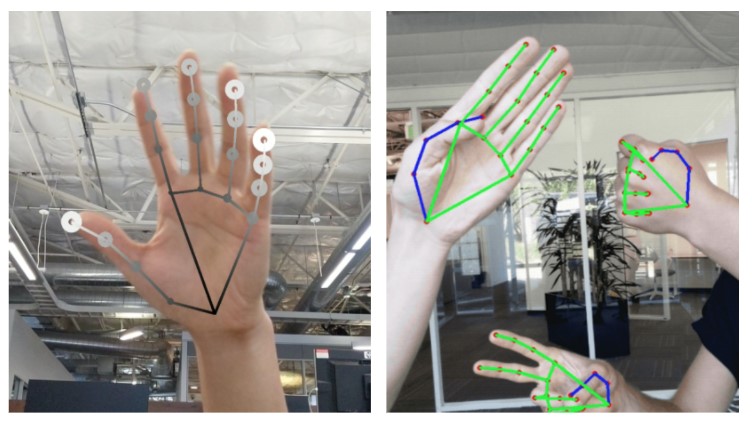
* 高效的两阶段手部跟踪管道，可在移动设备上实时跟踪多只手
* 仅凭 RGB 输入就能预测 2.5D 手部姿势的手部姿势估计模型.

图 1：渲染后的手部追踪结果。 (左图）： 以不同色调呈现的具有相对深度的手部地标。 圆圈越浅、越大，地标越靠近摄像头。 右图 Pixel 3 上的实时多手跟踪。图 1：渲染后的手部追踪结果。 (左图）： 以不同色调呈现的具有相对深度的手部地标。 圆圈越浅、越大，地标越靠近摄像头。 右图 Pixel 3 上的实时多手跟踪。

- 开源的手部跟踪管道可在各种平台上作为即用型解决方案使用，包括 Android、iOS、Web（Tensorflo-w.js[7]）和桌面 PC

# 架构

我们的手部追踪解决方案采用了由两个模型共同组成的 ML 管道：:

* - 手掌检测器可对完整输入图像进行操作，并通过定向手部边界框定位手掌位置.
* - 手部地标模型可在手掌检测器提供的裁剪手部边界框上运行，并返回高保真 2.5D 地标.

向手部地标模型提供精确裁剪的手掌图像大大减少了对数据增强（如旋转、平移和缩放）的需求，使网络能够将大部分能力用于提高地标定位的准确性。 在实时跟踪场景中，我们从上一帧的地标预测中推导出一个边界框，作为当前帧的输入，从而避免在每一帧上都应用检测器。 相反，只有在第一帧或手部预测显示手部丢失时，才会应用检测器。

## BlazePalm 检测器

为了检测最初的手部位置，我们采用了针对移动实时应用优化的单点检测器模型，该模型类似于 BlazeFace[1]，也可用于 MediaPipe[12]。 检测手的位置是一项非常复杂的任务：我们的模型必须能够检测到各种尺寸的手，同时具有较大的尺度跨度（∼20x），并且能够检测到被遮挡和自我遮挡的手。 脸部具有高对比度模式，例如眼睛和嘴巴周围，而手部则缺乏此类特征，因此很难仅从视觉特征可靠地检测到手。

我们的解决方案采用不同的策略应对上述挑战.

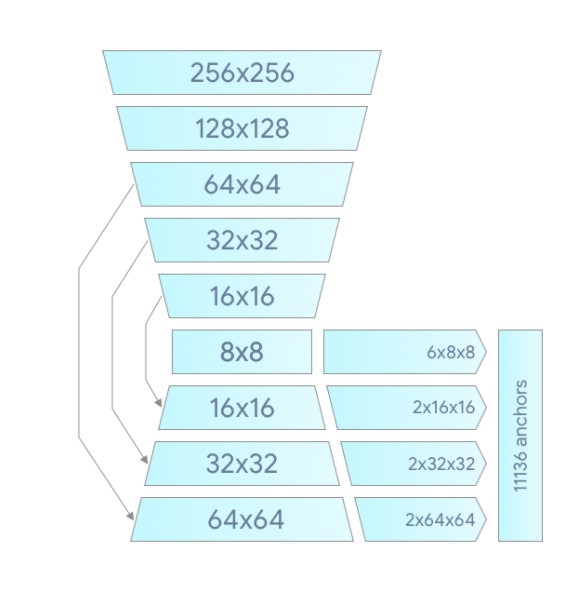
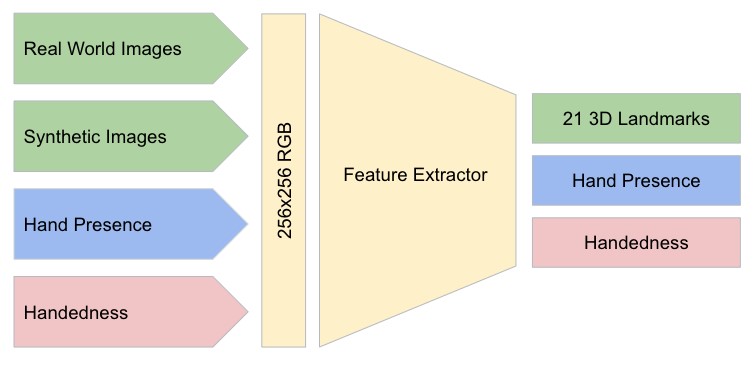
首先，我们训练的是手掌检测器而不是手部检测器，因为估算手掌和拳头等刚性物体的边界框要比检测带有关节手指的手简单得多。 此外，由于手掌是较小的物体，非最大抑制算法即使在双手自闭的情况下（如握手）也能很好地发挥作用。 此外，手掌可以只使用正方形边界框建模 [11]，忽略其他长宽比，因此可以将锚点数量减少 3∼5 倍。.

图 3：手部地标模型的结构。 该模型有三个输出，共享一个特征提取器。 每个头部都由标有相同颜色的相应数据集进行训练。 详见第 2.2 节。

图 2：手掌探测器模型结构。

其次，我们使用了与 FPN[9] 类似的编码器-解码器特征提取器，以获得更大的场景-上下文感知，即使是小物体也不例外。.

最后，我们在训练过程中尽量减少焦点损失[10]，以支持高尺度方差导致的大量锚点。 高级别手掌检测器架构如图 2 所示。 表 1 列出了对我们的设计元素进行的消融研究

## 手部地标模型

在对整个图像进行手掌检测后，我们的后续手部地标模型会通过回归对检测到的手部区域内的 21 个 2.5D 坐标进行精确的地标定位。 该模型能学习到一致的内部手部姿势表示，即使对于部分可见的手部和自我遮挡也很稳健。 该模型有三个输出（见图 3）:

1. 21 个手部地标，包括 x、y 和相对深度
2. 显示输入图像中手部出现概率的手部标志.
3. 手性的二元分类，如左手或右手。

我们对 21 个地标采用了与 [14] 相同的拓扑结构。 如下文所述，二维坐标是从真实世界图像和合成数据集中学习的，而手腕点的相对深度仅从合成图像中学习。 为了从追踪失败中恢复过来，我们开发了与 [8] 类似的另一种模型输出，用于产生在所提供的作物中确实存在合理对齐的手的概率。 如果得分低于阈值，就会触发检测器重置跟踪。 手感是在 AR/VR 中使用手进行有效交互的另一个重要属性。 这对于某些应用尤其有用，因为在这些应用中，每只手都与独特的功能相关联。 因此，我们开发了一种二元分类头，用于预测输入的手是左手还是右手。 我们的设置以移动 GPU 实时推理为目标，但我们也设计了更轻和更重的模型版本，以分别解决缺乏适当 GPU 支持的移动设备上的 CPU 推理和在桌面上运行时对准确性的更高要求。

# 数据集和注释

为了获得基本真实数据，我们针对问题的不同方面创建了以下数据集:

* - 野外数据集： 该数据集包含 6K 幅图像，种类繁多，如地理多样性、各种光照条件和手部外观。 该数据集的局限性在于不包含复杂的手部衔接。
* - 内部收集的手势数据集： 该数据集包含 10K 张图片，涵盖了所有物理上可能出现的手势的不同角度。 该数据集的局限性在于，它仅从 30 人中收集，且背景差异有限。 野外数据集和内部数据集互为补充，可提高鲁棒性
* - 合成数据集： 为了更好地覆盖可能的手部姿势并提供额外的深度监督，我们在各种背景上渲染了一个高质量的合成手部模型，并将其映射到相应的三维坐标上。 我们使用的是商用三维手部模型，该模型由 24 块骨骼组成，包含 36 个混合形状，可控制手指和手掌的粗细。 该模型还提供了 5 种不同肤色的纹理。 我们创建了手部姿势变换的视频序列，并从视频中采样 100K 张图像。 我们使用随机的高动态范围照明环境和三个不同的摄像头渲染了每个姿势。 示例见图 4。

对于手掌检测器，我们只使用野外数据集，该数据集足以定位手掌，并提供最丰富的外观。 不过，所有数据集都用于训练手部地标模型。 我们用 21 个地标对真实世界的图像进行注释，并使用投影的地面实况三维关节对合成图像进行注释。 对于手的存在，我们选择真实世界图像的一个子集作为正例，并在不包括注释手区域的区域作为负例进行采样。 对于手的状态，我们在现实世界的图像中标注了一个手的状态子集，以提供这样的数据.

# 结果

对于手部地标模型，我们的实验表明，将真实世界数据集和合成数据集结合使用能取得最佳效果。 详见表 2。 我们仅在真实世界图像上进行评估。 除了质量上的改进外，使用大型合成数据集进行训练还能减少视觉上的帧间抖动。 这一观察结果让我们相信，我们可以扩大真实世界数据集，以获得更好的泛化效果。

我们的目标是在移动设备上实现实时性能。 我们试验了不同大小的模型，发现 "全 "模型（见表 3）能很好地权衡质量和速度。 进一步增大模型容量只会带来微小的质量改善，但速度却会明显下降（详见表 3）。 我们使用 TensorFlow Lite GPU 后端进行设备上推理[6]。.

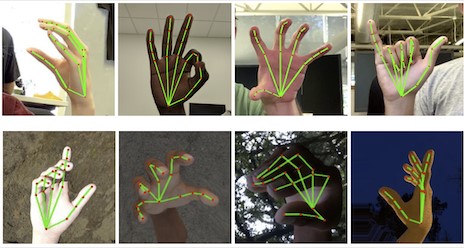


表 1：手掌探测器的烧蚀研究 手掌探测器的设计要素

图 4：我们的数据集示例。 (上图）： 经过注释的真实世界图像。 (下图）： 渲染后的合成手部图像，带有地面实况注释。 详见第 3 节。

表格

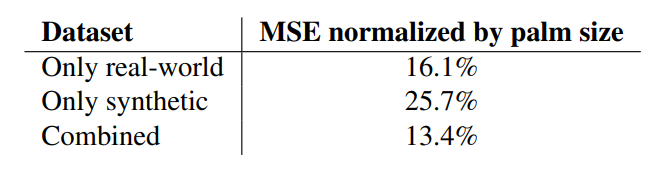
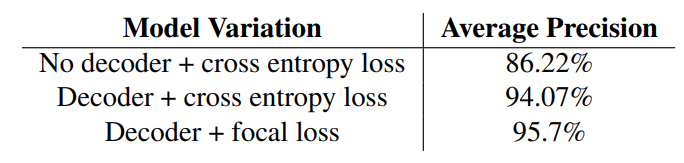
描述已自动生成

表 3：手部地标模型的性能特征。

表 2： 根据不同数据集训练出的模型结果

# 在 MediaPipe 中实施

有了 MediaPipe[12]，我们的手部追踪管道就可以构建成一个由模块化组件（称为计算器）组成的有向图。 Mediapipe 配备了一套可扩展的计算器，用于解决各种设备和平台上的模型推理、媒体处理和数据转换等任务。 对裁剪、渲染和神经网络计算等单个计算器进行了进一步优化，以利用 GPU 加速。 例如，我们在大多数现代手机上都采用了 TFLite GPU 推理技术。.

我们用于手部追踪的 MediaPipe 图如图 5 所示。 该图由两个子图组成，一个用于手掌检测，另一个用于地标计算。 MediaPipe 提供的一个关键优化功能是，手掌检测器只在需要时运行（相当不频繁），从而节省了大量计算量。 为此，我们从上一帧中计算出的手部地标推导出当前视频帧中的手部位置，从而无需在每一帧中应用手掌检测器。 为了提高鲁棒性，手部跟踪器模型还会输出一个额外的标量，用于捕捉输入作物中手部存在并合理对齐的置信度。 只有当置信度低于某个阈值时，手部检测模型才会重新应用于下一帧。

# 应用实例

我们的手部跟踪解决方案可随时用于手势识别和 AR 效果等许多应用中。 在预测的手部骨架基础上，我们采用一种简单的算法来计算手势，见图 6。 首先，通过累积的关节角度确定每个手指的状态，如弯曲或伸直。 然后，我们将一组手指状态映射到一组预定义的手势中。 这种简单而有效的技术使我们能够以合理的质量估算出基本的静态手势。 除了静态手势识别外，我们还可以使用一系列地标来预测动态手势。 另一种应用是在骨架上应用 AR 效果。 基于手部的 AR 效果目前非常流行。 在图 7 中，我们展示了一个霓虹灯风格的手部骨架 AR 渲染示例。.

# 结论

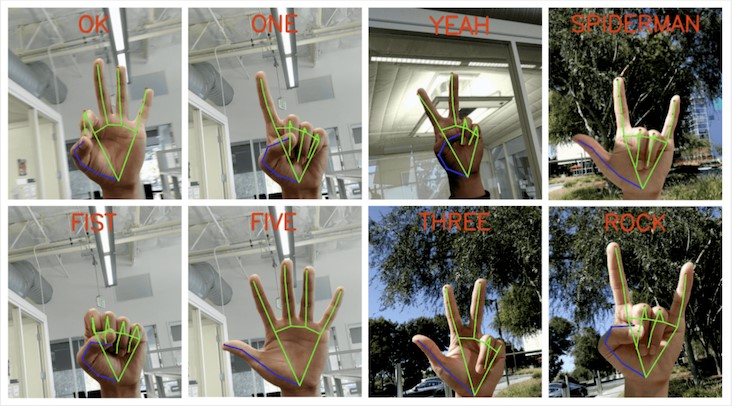
在本文中，我们提出了 MediaPipe Hands，这是一种端到端的手部跟踪解决方案，可在多个平台上实现实时性能。 我们的管道无需任何专用硬件即可预测 2.5D 地标，因此可轻松部署到商品设备上。 我们将该管道开源，以鼓励研究人员和工程师利用我们的管道构建手势控制和创意 AR/VR 应用。

图 6：实时手势识别截图。 手势的语义显示在图片顶部。

# References

1. Valentin Bazarevsky, Yury Kartynnik, Andrey Vakunov, Karthik Raveendran, and Matthias Grundmann. Blazeface: Sub-millisecond neural face detection on mobile gpus.

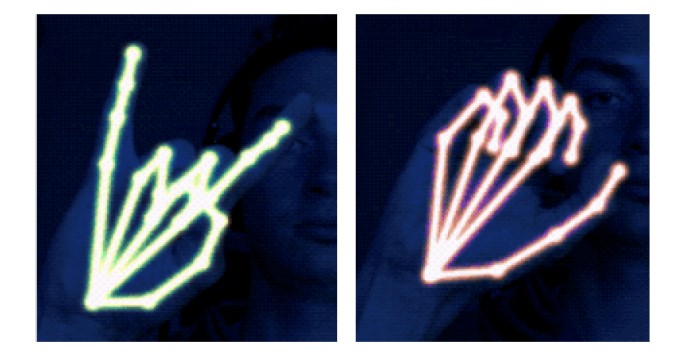
*CoRR*, abs/1907.05047, 2019. 2

图 7：基于我们预测的手部骨骼的实时 AR 效果示例

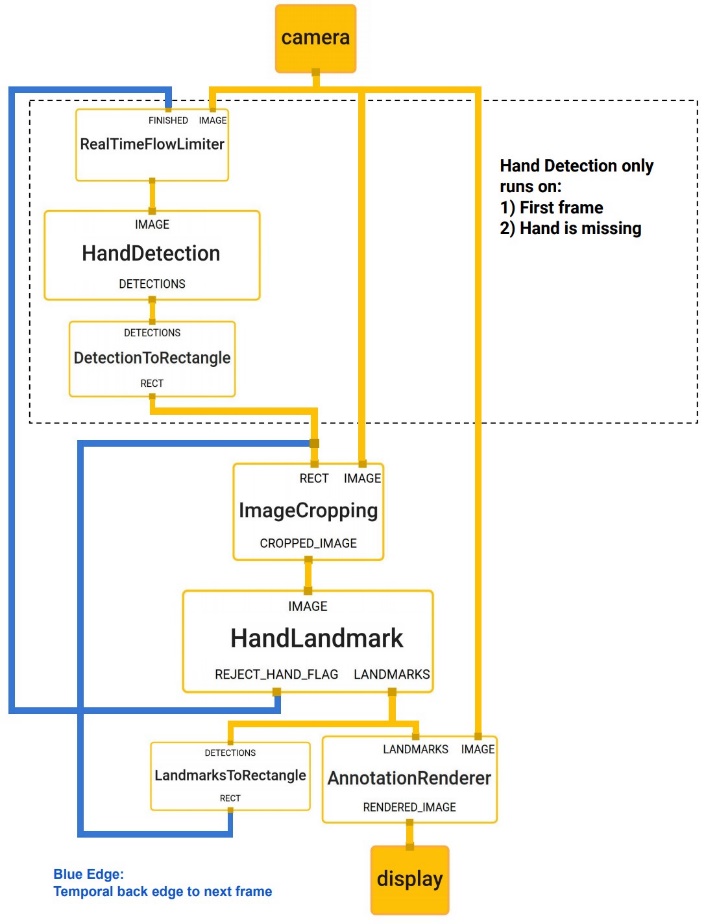
1. Facebook. Oculus Quest Hand Tracking. [https://www*.*oculus*.*com/blog/oculusconnect-6-introducing-hand-tracking-onoculus-quest-facebook-horizon-and-more/.](https://www.oculus.com/blog/oculus-connect-6-introducing-hand-tracking-on-oculus-quest-facebook-horizon-and-more/)

图 5：触发手部检测模型时，手部地标模型的输出控制。 这种行为是通过 MediaPipes 功能强大的同步构建模块实现的，从而为 ML 管道带来了高性能和最佳吞吐量。

1

1. Liuhao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann. Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3593–3601, 2016. 1
2. Liuhao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann. Robust 3d hand pose estimation from single depth images using multi-view cnns. *IEEE Transactions on Image Processing*, 27(9):4422–4436, 2018. 1
3. Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 3d hand shape and pose estimation from a single rgb image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 10833–10842, 2019. 1 [6] Google. Tensorflow lite on GPU.

[https://www*.*tensorflow*.*org/lite/ performance/gpuadvanced.](https://www.tensorflow.org/lite/performance/gpu_advanced) 3

1. Google. Tensorflow.js Handpose. [https://blog*.*tensorflow*.*org/2020/03/ face-and-hand-tracking-in-browser-withmediapipe-and-tensorflowjs*.*html.](https://blog.tensorflow.org/2020/03/face-and-hand-tracking-in-browser-with-mediapipe-and-tensorflowjs.html) 1
2. Yury Kartynnik, Artsiom Ablavatski, Ivan Grishchenko, and Matthias Grundmann. Real-time facial surface geometry from monocular video on mobile gpus. *CoRR*, abs/1907.06724, 2019. 2
3. Tsung-Yi Lin, Piotr Dollar, Ross B. Girshick, Kaiming He,´ Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. *CoRR*, abs/1612.03144, 2016. 2
4. Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollar.´ Focal loss for dense object detection. *CoRR*, abs/1708.02002, 2017. 2
5. Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexan-

der C. Berg. SSD: single shot multibox detector. *CoRR*, abs/1512.02325, 2015. 2

1. Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, ChuoLing Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. Mediapipe: A framework for building perception pipelines. volume abs/1906.08172, 2019. 1, 2, 4
2. Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Efficient model-based 3d tracking of hand articulations using kinect. 1
3. Tomas Simon, Hanbyul Joo, Iain A. Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. *CoRR*, abs/1704.07809, 2017. 2 [15] Snapchat. Lens Studio by Snap Inc. [https://lensstudio*.*snapchat*.*com/ templates/object/hand/.](https://lensstudio.snapchat.com/templates/object/hand/) 1
4. Andrea Tagliasacchi, Matthias Schroder, Anastasia Tkach,¨ Sofien Bouaziz, Mario Botsch, and Mark Pauly. Robust articulated-icp for real-time hand tracking. In *Computer Graphics Forum*, volume 34, pages 101–114. Wiley Online Library, 2015. 1
5. Chengde Wan, Thomas Probst, Luc Van Gool, and Angela Yao. Self-supervised 3d hand pose estimation through training by fitting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10853– 10862, 2019. 1