基于单幅图像的 3D 人体姿态估计结题报告

智能 16 唐超

1. 项目概述

人体姿态估计，pose estimation，就是通过将图片中已检测到的人体关键点正确的联系起来，从而估计人体姿态。人体关键点通常对应人体上有一定自由度的关节，比如颈、肩、肘、腕、腰、膝、踝等。通过对人体关键点在三维空间相对位置的计算，来估计人体当前的姿态。

基于单幅图像的 2D 人体姿态识别，目前发展已经比较成熟，例如 CMU 的 openpose、Facebook 的 densepose 等都已经能够实现准实时的 2D 人体姿态识别，然而在某些场景下，如体感游戏、行为识别等，仅提取 2D 的人体姿态信息是不够的，我们还试图从单幅图像中学习关节点的深度信息，并进一步得到 3D 的人体姿态估计结果，这就是本项目的目标：通过深度学习等方法，训练出一个可以识别 3D 人体姿态的模型，仅输入一副含有单人的野外图像给该模型，即可自动输出图像中人体的姿态估计结果。

传统的 3D 人体姿态估计问题，往往基于深度图或基于多目图像识别 3D 人体姿态，现有可用数据集也大多是 3D 的室内实验室图片或 2D 的野外（相对于实验室）场景图片，因此这一任务由于数据集的缺乏而极具挑战性。

2. 研究的主要内容

研究的内容分两个方面，前一部分时间主要用于实现 2D 的人体姿态识别，后面的时间主要用于 3D 的姿态识别。

2.1 对于 2D 的人体姿态识别

2.1.1 理论依据[1]

2.1.1.1 方法概述

人体姿态检测，通常是 top-down 的思路，也就是先做行人检测，然后把每一个人分割出来，最后基于每一个独立个体，找出各自的人体关键点。这个办法有两个问题：

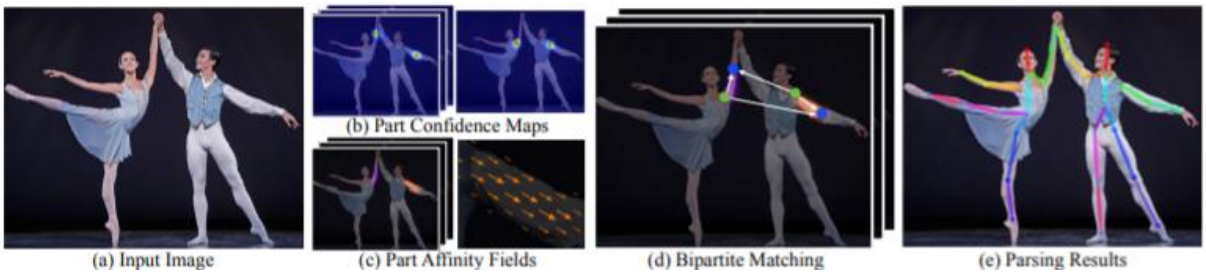
- 结果严重依赖第一步行人检测器的结果，如果人都没找到，就无从找到人体关键点了。
- 计算时间和人数正相关，人越多越耗费时间。

OpenPose 使用了另一种思路，即 bottom-up，先找出图中所有的关键点，再用匹配的方法拼装成一个个人体骨架。这种办法有一个缺陷，就是没办法利用全局上下文的信息。为了克服这个问题，本文想出了一个办法，就是 PAF（Part Affinity Fields)，部分区

域亲和。它负责在图像域编码着关键点位置和方向的 2D 矢量。同时，使用 CMP（Part Detection Confidence Maps）标记每一个关键点的置信度（就是常说的“热图”）。通过两个分支，联合学习关键点位置和他们之间的联系。同时推断这些自下而上的检测和关联的方式，利用贪婪分析算法（Greedy parsing Algorithm），能够对全局上下文进行足够的编码，获得高质量的结果，而只是消耗了一小部分计算成本。并行情况下基本达到实时，且耗时与图片中的人数无强关联。

图 1 部分区域亲和 PAFs

整体流程：(a)输入图像 --> (b)预测关键点置信度 & (c)关键点亲和度向量 --> (d)关键点聚类 --> (e)骨架组装。如图二所示。



2.1.1.2 网络结构

首先，输入的图像将经过 10 层 VGG-19 网络转化为图像特征 F，再分成两个分支分别迭代预测图像关键点置信度（热图，用 S 表示）和区域亲和力向量场（用 L 表示），

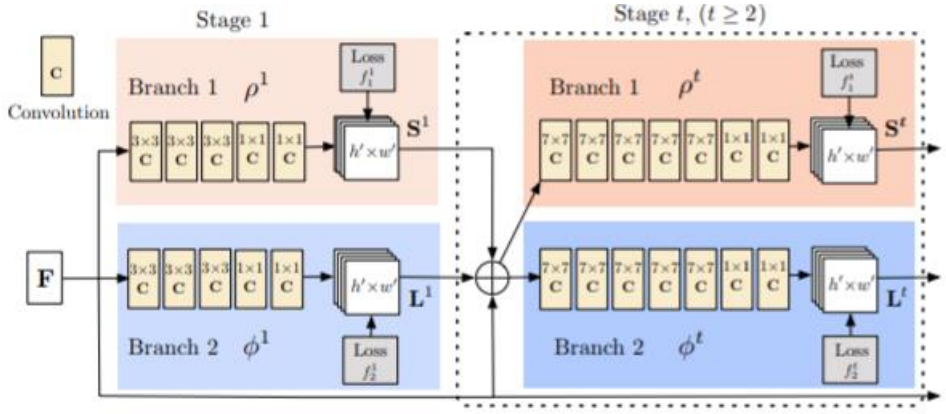


图 2 网络结构

第一次预测的输入仅为图像特征 F，$S^1 = \rho^1(F), L^1 = \Phi^1(F)$，以后的每一次迭代的输入为上一次预测的关键点置信度、区域亲和力向量场及最初的图像特征 F.

$$\mathbf{S}^t = \rho^t(\mathbf{F}, \mathbf{S}^{t-1}, \mathbf{L}^{t-1}), \ \forall t \geq 2$$
$$\mathbf{L}^t = \phi^t(\mathbf{F}, \mathbf{S}^{t-1}, \mathbf{L}^{t-1}), \ \forall t \geq 2$$

对于每一个 stage 的损失函数定义如下：

$$f_{\mathbf{S}}^t = \sum_{j=1}^{J}\sum_{\mathbf{p}} \mathbf{W}(\mathbf{p}) \cdot \|\mathbf{S}_j^t(\mathbf{p}) - \mathbf{S}_j^*(\mathbf{p})\|_2^2$$

$$f_{\mathbf{L}}^t = \sum_{c=1}^{C}\sum_{\mathbf{p}} \mathbf{W}(\mathbf{p}) \cdot \|\mathbf{L}_c^t(\mathbf{p}) - \mathbf{L}_c^*(\mathbf{p})\|_2^2$$

其中， p 表示图像中的每一个像素点， $S_j^*(p)$， $L_c^*(p)$ 分别为每个点的关键点置信度和亲和力向量的 groundtruth，对于在数据集中可能存在关键点标注缺失的情况，引入 $W(p)$，如果标注缺失， $W(p)=0$，其他情况， $W(p)=1$.

整体的目标方程为：

$$f = \sum_{t=1}^{T}(f_{\mathbf{S}}^t + f_{\mathbf{L}}^t)$$

2.1.1.3 关键点置信度

图像中每个点对其中第 k 个人的第 j 个 groundtruth 关键点的置信度定义如下：

$$\mathbf{S}_{j,k}^*(\mathbf{p}) = \exp\left(-\frac{\|\mathbf{p} - \mathbf{x}_{j,k}\|_2^2}{\sigma^2}\right)$$

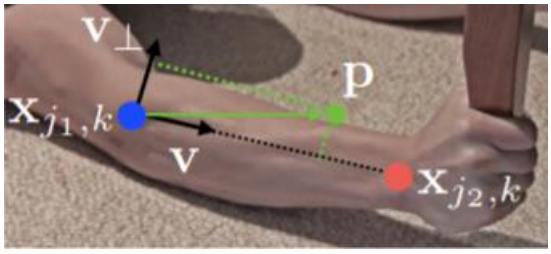
当图中有多个人时，每个点的置信度取不同人影响下的最大值即可：

$$\mathbf{S}_j^*(\mathbf{p}) = \max_k \mathbf{S}_{j,k}^*(\mathbf{p})$$

在测试时，对于置信度网络的输出直接采用 NMS（非极大值抑制）获取关键点位置。

2.1.1.4 区域亲和向量场

假设 $\mathrm{x}_{j_1,k}, \mathrm{x}_{j_2,k}$ 分别为图像中第 k 个人的第 c 个 limb 的 groundtruth 关键点 j_1, j_2 的位置， v 为从 j_1 到 j_2 的单位向量， $\mathbf{v} = (\mathbf{x}_{j_2,k} - \mathbf{x}_{j_1,k})/\|\mathbf{x}_{j_2,k} - \mathbf{x}_{j_1,k}\|_2$.



如果点 p 在第 c 个 limb 上，即满足：

$$0 \le \mathbf{v} \cdot (\mathbf{p} - \mathbf{x}_{j_1,k}) \le l_{c,k} \text{ and } |\mathbf{v}_\perp \cdot (\mathbf{p} - \mathbf{x}_{j_1,k})| \le \sigma_l$$

则 $L_{c,k}^*(p) = v$， 否则 $L_{c,k}^* = 0$. 对于多个人，

$$\mathbf{L}_c^*(\mathbf{p}) = \frac{1}{n_c(\mathbf{p})} \sum_k \mathbf{L}_{c,k}^*(\mathbf{p})$$

其中 $n_c(p)$ 为图像中在 p 点亲和力向量不为 0 的人数。

在测试时，对于可能相关联的两关键点 d_{j_1} 和 d_{j_2}，其亲和度定义 $L(p)$ 在两点连线上的线积分：

$$E = \int_{u=0}^{u=1} \mathbf{L}_c(\mathbf{p}(u)) \cdot \frac{\mathbf{d}_{j_2} - \mathbf{d}_{j_1}}{||\mathbf{d}_{j_2} - \mathbf{d}_{j_1}||_2} du$$

其中 $p(u) = (1-u)d_{j_1} + ud_{j_2}$.

2.1.1.5 关键点聚类

预测了关键点置信度后，用 NMS 方法获取关键点位置，构成关键点集:

$D_J = \{d_j^m \mid j \in \{1...J\}, m \in \{1...N_j\}\}$ ，其中 d_j^m 表示第 j 类关键点的第 m 个候选点，N_j 为第 j 个关键点的可能候选点的数目。定义 $z_{j_1j_2}^{mn} \in \{0,1\}$ 表示两个关键点 $d_{j_1}^m$ 与 $d_{j_2}^n$ 是否应该连线（是否时同一个人同一肢体的两端），值为 1 表示应该连线，值为 0 表示不应该连线。所有关键点间的连线情况构成集合 Z：

$$Z = \{z_{j_1j_2}^{mn} \mid j_1, j_2 \in \{1...J\}, m \in \{1...N_j\}, n \in \{1...N_{j_2}\}\}$$

对于给定的肢体 c 与其相关联的两类关键点是 j1, j2, 这两类关键点的聚类问题可以下列最优化问题

$$\max_{Z_c} E_c = \max \sum_{m \in D_{j_1}} \sum_{n \in D_{j_2}} E_{mn} \cdot z_{j_1j_2}^{mn}$$

$$s.t. \begin{cases} \forall m \in D_{j_1}, \sum_{n \in D_{j_2}} z_{j_1j_2}^{mn} \le 1 \\ \forall n \in D_{j_2}, \sum_{m \in D_{j1}} z_{j_1j_2}^{mn} \le 1 \end{cases}$$

其中 E_{mn} 是 $d_{j_1}^m$ 与 $d_{j_2}^n$ 的亲和度，E_c 是肢体 c 上可能关联的两类关键点间的连线总亲和度。该问题刻用匈牙利算法求解。

对于所有肢体，关键点连线聚类问题可以看成：

$$\max_Z \mathrm{E} = \sum_{c=1}^{C} \max_{Z_c} E_c$$

也就是各肢体之间独立优化配对,解决了每个肢体涉及的两类关键点的连线聚类以后,最后依据关键点相同衔接组装成整个身体的姿态。

2.1.2 测试效果及分析

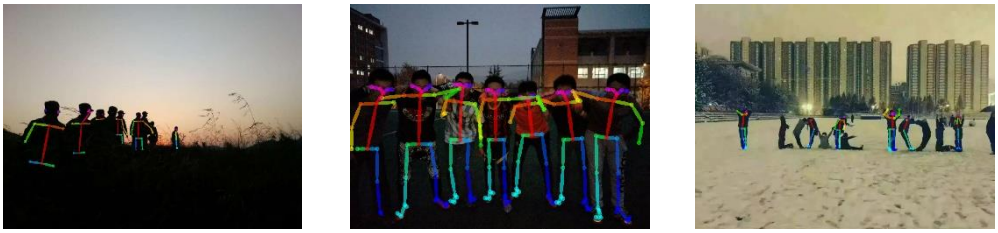
对各种可能人体姿态做 2D 姿态识别测试，结果如下。

单人的站、坐、趴、躺、俯卧撑等姿势测试结果如图 所示，可以看到遮挡对关键点的估计造成了一定的影响，如（a）中眼睛（紫色点）的位置估计错误，（c）中右手小臂

没有测出来，（d）中盆骨的位置明显有估计偏差。另外，在（c）中当眼睛闭上时，把眼镜错误估计为了眼睛，可能是训练集中对于这样的“极端情况”涉及较少的原因，不过整体的姿态估计结果已经非常不错了。

(a) (b) (c) (d) (e)

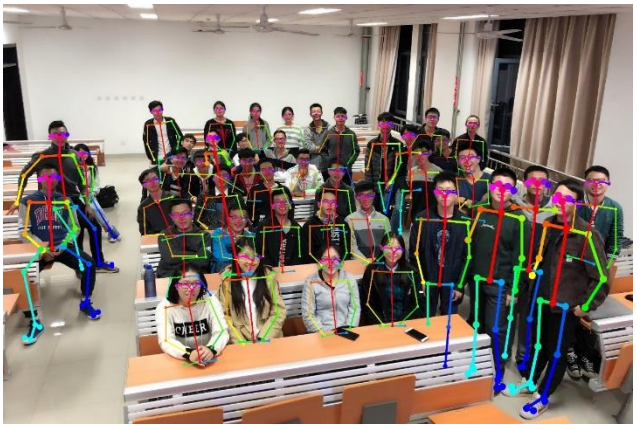
对于光线不足的情况也有非常不错的识别效果。



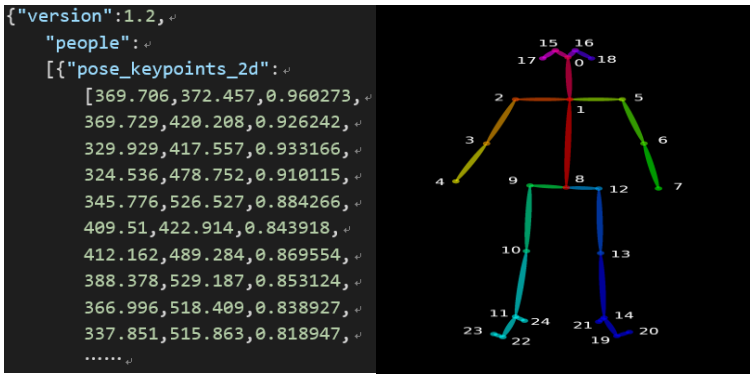
对于非真实人类的测试结果如图 所示，可以看到对于油画中的人物(a)(e)都有不错的检测效果，对于非人类的类似人的动作(b)(c)也有不错的泛化能力，但过强的泛化能力也导致了(c)中雕塑的姿态估计出现了错误，将雕塑下方的几条凹槽识别为了人。

(a) (b) (c) (d) (e)

对于多人的姿态估计，openpose 的最大特点在于其识别时间几乎不随人数的增多而增大，如下方的图像为在同样的时间内得到的估计结果。可以看出，在得到一个人体尽量多的语义信息情况下，对于关键点的部分遮挡，也有着不错的识别结果。



可以对每个关键点的坐标及置信度以 json 格式文件输出，每个关键点信息按顺序对应 body_25 数据集的关键点位置，如图 （左边关键点信息来自图 (b)的部分姿态估计结果），这一数据可以作为 3D 姿态估计或其他应用的输入。



另外，openpose 还可以做实时的人体姿态识别及人脸、手的识别。

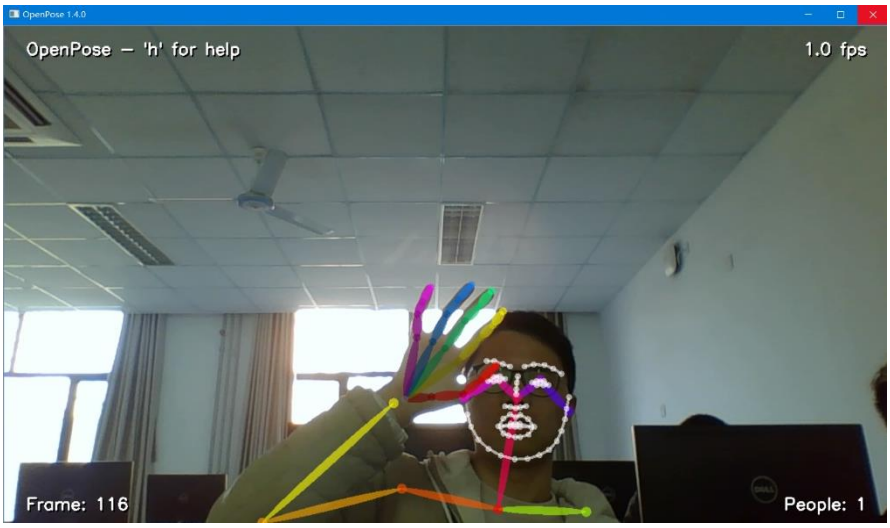


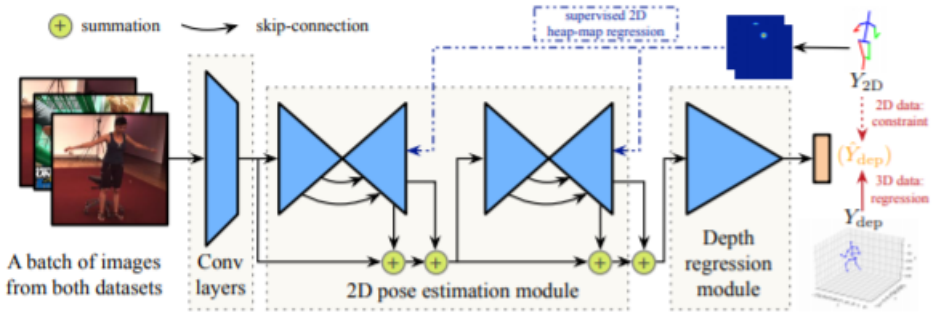
图 2 实时身体、脸部、手部识别

2.2 3D 的人体姿态识别

对于 3D 的人体姿态识别，主要学习了复旦大学周星壹的开源项目 pose-hg-3d。该项目基于 MPII、Human3.6M 等人体姿态数据集、深度学习框架 pytorch，在野外单人 3D 姿态估计上取得了很好的效果。

2.2.1 方法概述[2]

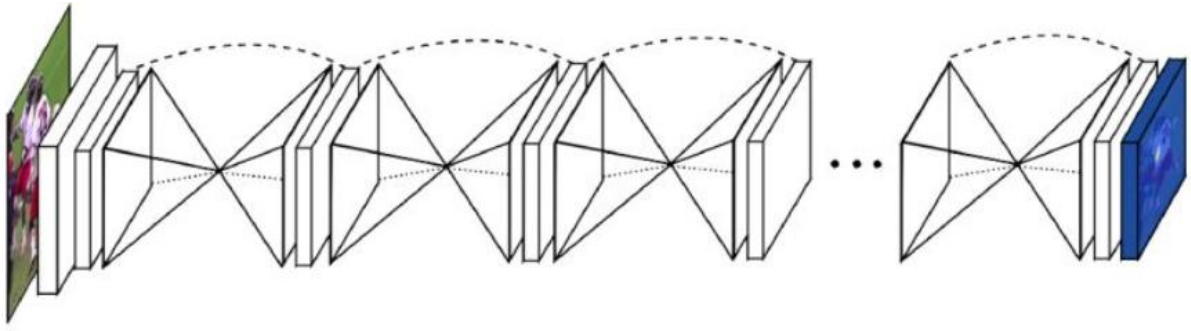
与以往 3D 人体姿态估计所采用的方法不同的是，pose-hg-3d 不再将检测 2D 关节点与 3D 骨架回归分离，而是将两个网络用中间层连接为一个连续的网络进行训练。在测试中，图像通过堆积的沙漏网络转换为 2D 热图，将 2D 热图和具有低层图像特征的热图汇总作为深度回归模块的输入。在训练中，来自 2D 和 3D 数据集的图像被混合在一批处理中。对于 3 维数据，采用欧几里得损失的标准回归，对于二维数据，其损失函数基于二维标注和人体骨骼先验知识的弱监督。



2.2.1.1 堆叠沙漏网络[3]

以前估计姿态的网络结构，大多只使用最后一层的卷积特征，这样会造成信息的丢失。事实上，对于姿态估计这种关联型任务，全身不同的关节点，并不是在相同的 feature map 上具有最好的识别精度。举例来说，胳膊可能在第 3 层的 feature map 上容易识别，而头部在第 5 层上更容易识别。所以，需要设计一种可以同时使用多个 feature map 的网络结构。

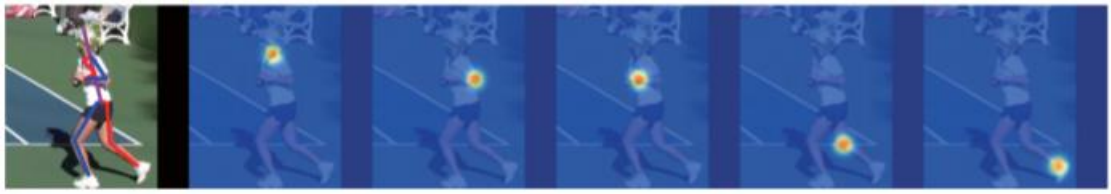
Stacked Hourglass Networks 网络结构能够捕获并整合图像所有尺度的信息。之所以称这种网络为 Stacked Hourglass Networks，主要是它长得很像堆叠起来的沙漏，如下图所示。



关节点之间是可以互相参考预测的，即知道双肩的位置后，可以更好的预测肘部节点，给出腰部和脚踝位置，又可以用于预测膝盖。既然热图代表了输入对象的所有关节点，那么热图就包含了所有关节点的相互关系，可以看作是图模型。所以将第一个沙漏网络给出的热图作为下一个沙漏网络的输入，就意味着第二个沙漏网络可以使用关节点件的相互关系，从而提升了关节点的预测精度。

这种堆叠在一起的 Hourglass 模块结构是对称的，卷积和 max pooling 被用来将特征降到一个很低的分辨率，在每一个 max pooling 步骤中，网络产生分支并在原来提前池化的分辨率下使用更多的卷积，当到达最低的分辨率的时候，网络开始 upsample 并结合不同尺度下的特征。这里 upsample（上采样）采用的方法是最近临插值，之后再将两个特征集按元素位置相加。

当到达输出分辨率的时候，再接两个 1×1 的卷积层来进行最后的预测，网络的输出是一组热图，对于给定的热图，网络预测在每个像素处存在关节的概率。



2.2.1.2 深度回归模块

在训练中同时采用了二维标注和三维标注的人体姿态数据集，在进行深度回归时，对于完全标注的 3D 数据集 $S_{3D}=\{I_{3D},y_{2D},y_{dep}\}$，使用 groundtruth depth label，训练损失可以简单地表示为标准的欧几里得损失。对于弱标记数据集 $S_{2D}=\{I_{2D},y_{2D}\}$，采用一种基于骨长几何约束作为训练损失。总的来说，用 $\hat{Y}_{dep}$ 表示预测深度，深度回归模块的损失函数为

$$L_{dep}(\hat{Y}_{dep}\mid I,Y_{2D})=\begin{cases}\lambda_{reg}\parallel Y_{dep}-\hat{Y}_{dep}\parallel^2, I\in I_{3D}\\ \lambda_{geo}L_{geo}(\hat{Y}_{dep}\mid Y_{2D}), I\in I_{2D}\end{cases}$$

其中，λ_{reg}，λ_{geo} 分别为对应的损失权重，$L_{geo}(\hat{Y}_{dep}\mid Y_{2D})$ 为基于人体骨骼骨长比保持相对固定这一事实（例如大臂与小臂的骨长比固定，左肩与右肩的骨长相同）所计算的几何损失。

定义以下四组骨骼：R_{arm}={left lower/upper arm, right lower/upper arm}，R_{leg}={left lower/upper leg, right lower/upper leg}，$R_{shoulder}$={left shoulder bone, right shoulder bone}，R_{hip}={left hip bone, right hip bone}. 对每组骨骼 R_i 中的每类骨 e，用 l_e 表示骨 e 的长度，$\overline{l}_e$ 表示全体数据集中骨 e 长度的均值，$\frac{l_e}{\overline{l}_e}$ 应保持固定，进而 $L_{geo}(\hat{Y}_{dep}\mid Y_{2D})$ 可以表示为 $\{\frac{l_e}{\overline{l}_e}\}_e\in R_i$ 的方差和：

$$L_{geo}(\hat{Y}_{dep}\mid Y_{2D})=\sum_i\frac{1}{\mid R_i\mid}\sum_{e\in R_i}\left(\frac{l_e}{\overline{l}_e}-\overline{r}_e\right)^2$$

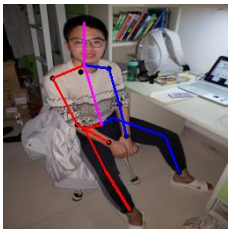
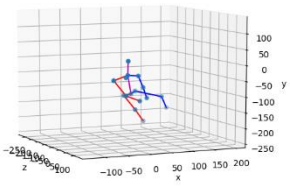
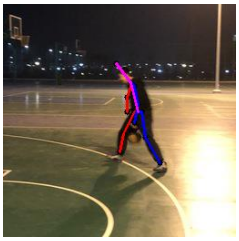
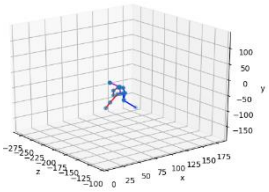
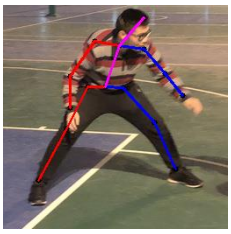
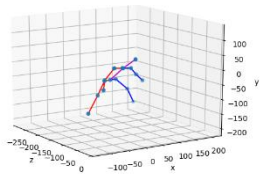
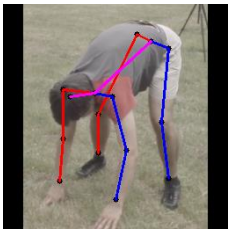
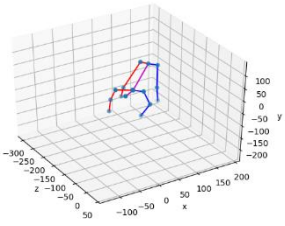
其中

$$\overline{r}_i=\frac{1}{\mid R_i\mid}\sum_{e\in R_i}\frac{l_e}{\overline{l}_e}$$

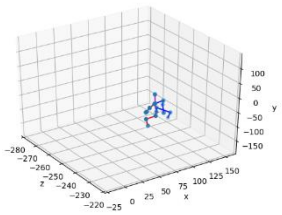
注意到骨长是关节点位置的函数，而关节点位置又是预测深度的函数，因此 L_{geo} 是关于 $\hat{Y}_{dep}$ 的连续可微的函数。

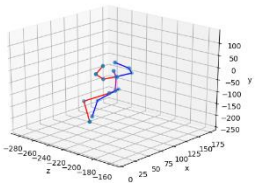
2.2.2 测试效果及分析

采用多种图像进行 3D 人体姿态估计测试，在没有复杂背景的单人图像上取得了不错的效果，如下图所示。

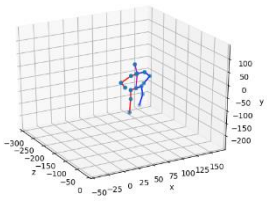









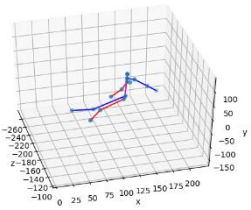
在进行更多的测试过程中发现，在某些膝盖或肘部没有弯曲的情况下，会将关节点深度值回归错误，造成 3D 姿态估计结果“扭曲”。

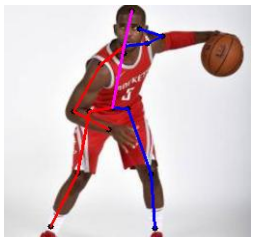
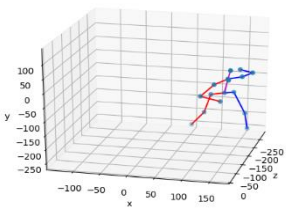


接着试图在多人的图像中进行测试，模型会选择位于图像正中间的人进行姿态估计，在这种有干扰的情况下，往往在检测某个人的 2D 姿态这一步就出了问题，更不用说 3D 姿态估计了。



另外，在一些单人简单背景下，其姿态估计结果也不尽如人意，如图 ，可以看到，由于篮球和红色护臂的遮挡，左手的二维姿态就估计错了。

尝试对前述 2D 人体姿态估计错误的测试图用 OpenPose 重新估计 2D 姿态，结果如下所

示，可以看出这一估计结果明显好于原来的结果。


/*虽然在 pose-hg-3d 项目中，作者提到将 3D 姿态估计分割为 2D 关节点位置估计与 3D 关节点位置回归两个独立子任务的方法会造成在后一任务中图像语义信息的丢失，但明显此项目的 2D 姿态估计结果的偏差已经对其最终结果造成了极大的影响，*/因此，采用更好的 2D 姿态估计方式，如 OpenPose，也许在一定程度上会对最终的 3D 姿态估计结果产生积极的影响。

总得来说，这一项目在野外单人图像上有着不错的 3D 姿态估计效果，但不能推广到更复杂的测试场景。

3. 研究过程及反思

由于之前对深度学习几乎没有了解，也从没有复现过别人的论文或测试过别人的开源项目，甚至连 git 也不会用、命令行也看不懂，在项目开题时就陷入了困境，开源项目的 github 中 Pytorch、torch7、torsorflow、caffe、CUDA、cudnn、Ubuntu……都看得我一头雾水，十几页的描述神经网络结构的英文论文更不敢看，后来大部分时间都用在了装环境和跑别人的项目上，做 2D 姿态估计时，除了 openpose，还尝试过 Human Body Pose Estimation、Convolutional Pose Machines 等项目，虽然一直在踩坑，最终只对别人的项目做了测试，并没有做自己的东西，但在这一过程中学习了两篇论文、git 的使用、Ubuntu 双系统，了解了一些网络模型、Pytorch 和 tensorflow 等深度学习框架、一些以前没见过的文件格等，并积累了各种装环境的经验，也算有所收获。

4. 参考论文及开源项目

4.1 参考论文

[1]Cao Z, Simon T, Wei S E, et al. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields[J]. 2016.

[2] Zhou X, Huang Q, Xiao S, et al. Towards 3D Human Pose Estimation in the Wild: A Weakly-Supervised Approach[J]. 2017.

[3] Newell A , Yang K , Deng J . Stacked Hourglass Networks for Human Pose Estimation[J]. 2016.

4.2 开源项目

OpenPose: https://github.com/CMU-Perceptual-Computing-Lab/openpose

pose-hg-3d: https://github.com/xingyizhou/pytorch-pose-hg-3d