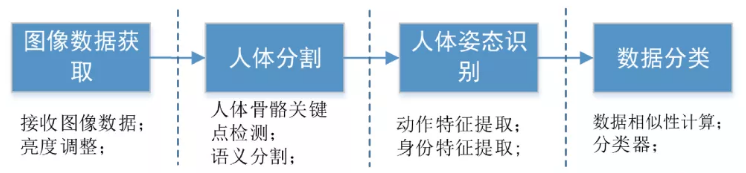
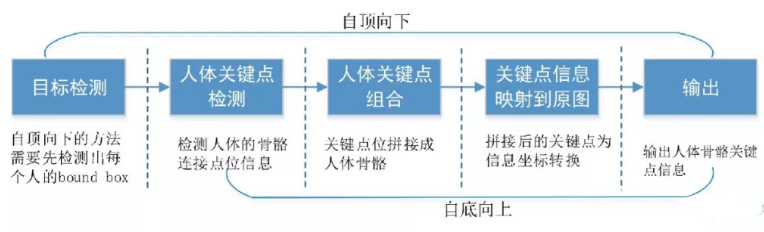
###### 姿态识别简介

**姿态识别一般流程如下：**



**人体分割**

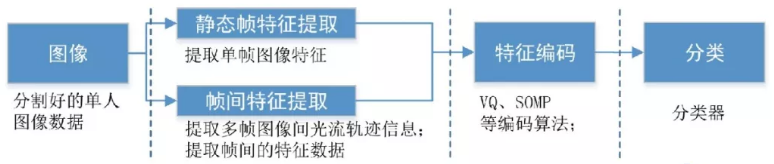
人体分割使用的方法可以大体分为人体骨骼关键点检测、语义分割等方式实现。这里主要分析与姿态相关的人体骨骼关键点检测。人体骨骼关键点检测输出是人体的骨架信息，一般主要作为人体姿态识别的基础部分，主要用于分割、对齐等。一般实现流程为：



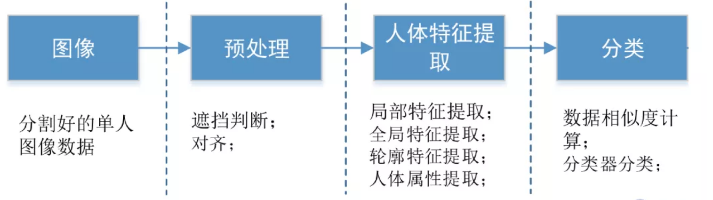
**人体姿态识别**

人体姿态识别包括动作识别、身份识别两个方面，关键在人体特征提取，人体特征提取主要完成动作特征提取、身份特征提取。一般实现流程为：

**动作识别**



**身份识别**



人体骨骼关键点检测常常用来作为姿态识别的基础部件，下面介绍人体骨骼关键点检测实现。

###### **人体骨骼关键点检测**

人体骨骼关键点检测也称为姿态估计（Pose Estimation），主要检测人体的关键点信息，如关节，五官等，通过关键点描述人体骨骼信息，常用来作为姿态识别、行为分析等的基础部件，如下图所示：

###### 

* 1. 人体骨骼关键点检测现状

人体骨骼关键点检测是一种多方面任务，包含了目标检测、人体骨骼关键点检测、分割等。人体骨骼关键点检测可以分为二维（2D）和三维（3D）的人体骨骼关键点检测；按照检测的方法又能分为自底向上、自顶向下两个方式。

人体骨骼关键点检测的挑战：

1. 每张图片中包含人的数量是未知的，图像中人越多，计算复杂度越大（计算量与人的数量正相关），这使得处理时间变长，从而使real time变得困难。

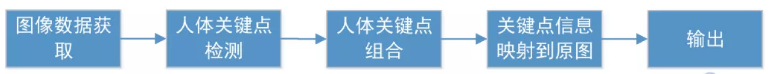
2. 人与人之间会存在如接触、遮挡等关系，导致将不同人的关键节点区分出来的难度增加，有可能会将骨骼关键点误认为是另一个人的。

3. 关键点区域的图像信息比较难区分，也就是说某个关键点检测时容易出现检测位置不准或者置信度不准，甚至将背景的图像当成关键点图像的错误。

4. 人体不同关键点检测的难易程度是不一样的，对于腰部、腿部这类没有比较明显特征关键点的检测要难于头部附近关键点的检测，需要对不同的关键点区别对待。

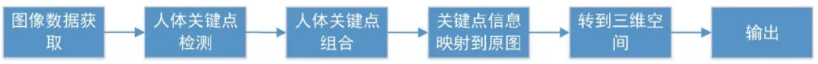
根据输出的关键点数据是2D还是3D的，分为2D人体骨骼关键点检测、3D人体骨骼关键点检测。

**2D人体骨骼关键点检测：**



输入的图像数据：RGB

**3D人体骨骼关键点检测：**



输入的图像数据：RGB（2D+）

输入的图像数据：RGBD（3D）

* 1. 2D人体骨骼关键点检测实现

2D人体骨骼关键点检测的算法模型基本上都遵循自顶向下（TopDown）、自底向上（BottomUp）两种思路来实现：



**自顶向下**

自顶向下（Topdown）的人体骨骼关键点检测算法包含两个部分：目标检测和单人人体骨骼关键点检测，先检测人体，再做单人姿态估计的两阶段方法。(G-RMI, RMPE, CPN, SimpleBaseline,HRNet[, ...)

存在的问题

* 必然受到了目标检测任务的制约。
* 基于bounding box的单人姿态估计问题，在面对遮挡问题容易受到挫折。
* 精度虽然髙实时性能较差，运行推断时间runtime和检测的人数成正比关系
* 小尺寸图像受限

**自底向上**

自底向上（BottomUp）的人体骨骼关键点检测算法主要包含两个部分：关键点检测和关键点聚类连接，关键点检测目的是将图片中所有人的所有关键点全部检测出来。关键点检测完毕之后需要对这些关键点进行聚类处理，将每一个人的不同关键点连接在一块，从而连接产生不同的个体。

存在的问题

* 精度不如Top-down的更加精准，但实时性能较好
* bounding box free
* 面对拥挤问题、遮挡问题仍然容易受到挫折
* 图像上的人体的尺度大小，未经归一化，分布很不均匀，关键点特征的提取难大于Top-down的方法

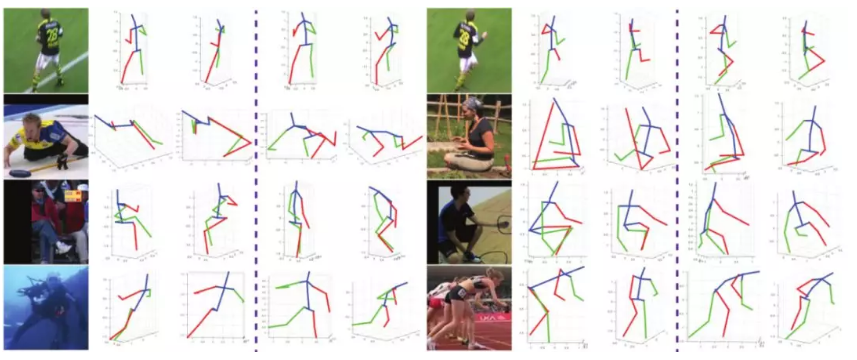
对于2D多人姿态估计来讲，其难题仍然是在互遮挡、自遮挡、复杂背景环境干扰。2D 关键点预测是姿态估计任务的基础性研究问题，但它并不是代表着姿态估计就仅仅是预测关键点，其本身难以克服的问题如拥挤、自遮挡或相互遮挡，就是因为深度、3D信息的缺乏而导致，也许只有上升到3D的层面时才能被解决。

当需要3D的人体姿态信息时，2D人体骨骼关键点检测并不能满足要求，这时需要3D人体骨骼关键点检测。

* 1. 2D+人体骨骼关键点检测实现

2D+人体骨骼关键点检测多使用3D重建的方法，一般输入都是RGB的2D图像数据，首先使用2D人体骨骼关键点检测的方法检测出2D的人体骨骼关键点后，再使用3D重建的方式，将2D的关键点信息转到3D的空间中，输出的就是3维人体骨骼关键点信息。这种方法使用的图像数据不是真的3D数据，输出的3维的信息主要是通过2D到3D重建的方式实现的，当然这种方法也能解决一定的问题，在动作识别、身份识别等领域都有一定的作用。

2D+人体骨骼关键点检测输出的人体骨骼关键点信息是3D数据，即每个关节都是一个3D坐标（x,y,z），如下图所示：



2D+人体骨骼关键点检测的模型速度比较快的是DensePose

DensePose-RCNN采用的是金字塔网络（FPN）特征的RCNN结构，区域特征聚集方式ROI align pooling以获得每个选定区域内的密集部分标签和坐标。将2D图像中人的表面图像数据投影到3D人体表面上，将人体的3D表面模型切分为24个部分，然后为每一部分构建一个UV坐标系，将2D图像上的人体部分的每一个点映射到相应的3D表面部分。也可以在估计出图像中人体的UV之后，将3Dmodel通过变换，将空间坐标转换为UV坐标之后，贴到图像上。

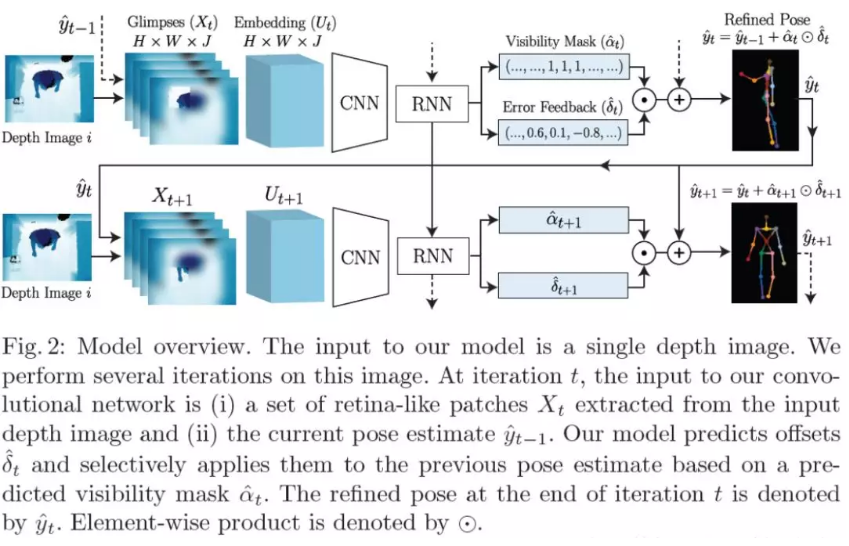
DensePose借鉴了Mask-RCNN的结构，同时带有Feature Pyramid Network（FPN）的特征，以及ROI-Align池化。整体流程为：首先使用Faster-RCNN得到人物区域bounding box，然后使用CNN网络模块分块，分块后用CNN网络模型处理每一个分块，最后得到目标的热力图IVU。

* 1. 3D人体骨骼关键点检测实现

前面分析的2D+人体骨骼关键点检测，模式上基本能总结为2D的人体骨骼关键点检测+3D重建的过程。这里分析使用RGBD图像数据的3D人体骨骼关键点检测实现。

图像数据是RGBD的一种方式就是使用深度图像来获取人体骨骼关键点数据，虽然使用的只是RGBD中的深度数据D（Depth Image），但是Depth Image实际上也是一幅真实的3D图像数据，Depth Image包含了深度信息、图像的高宽信息。很多硬件设备的SDK都带有该功能，如微软的Kinect、英特尔实感3D摄像头、Mo-Cap解决方案等。

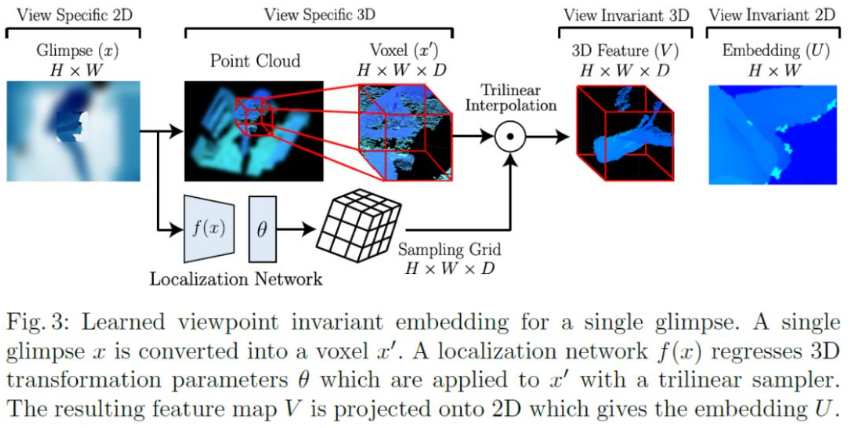
下面分析基于CNN+RNN结构的人体骨骼关键点获取模型，模型的网络结构如下：



该模型的结构如上图所示，输入一张深度图，对深度图片进行多次迭代处理，每一个迭代包含两个输入，即原图处理生成的一个patch（称为Glimpses）和之前生成的人体骨骼关键点信息。

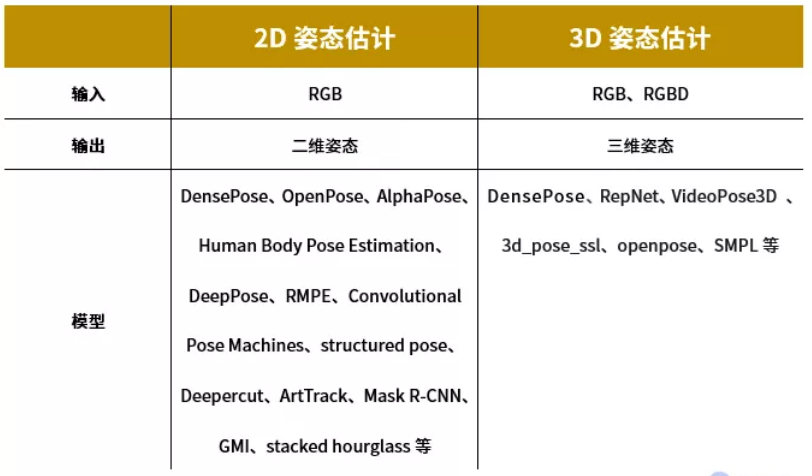
该模型的输入深度图片并不直接输送到网络中去，而是先通过提取图片中的不同的patch，其中每个patch都以每个预测的身体部位为中心，并且将其转化成为Glimpses。Glimpse是一种对原始输入的特殊编码形式，它的中心分辨率高，周围分辨率逐渐降低，这就使得中心的特征得到重点的学习，同时又保留了部分空间信息。

上图的Depth Image到Embedding的过程输出的是具有视角不变特性的特征空间，获取到了尺度不变的Glimpses之后，将Glimpses作为输入，经过运算转换成3维的姿态数据输出，转换模型如下：



对于一个输入的Glimpse，结合原来的深度数据经过函数f(x)转换成体素（3维），最后将3D特征V并映射回到二维空间，最后得出3维的人体姿态信息。

2D的人体骨骼关键点检测跟3D的人体骨骼关键点检测对比如下



###### **人体姿态识别**

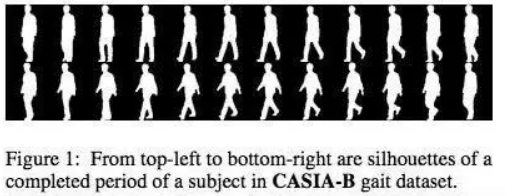
* 1. 动作识别

视频逐帧分析，采用连续的动作识别出人物动作，如走路、跑步、蹲下等。在计算机视觉中人机交互中有很大的应用，主要处理模型大概分为两个大类：卷积神经网络（3D-CNN）、基于循环神经网络与其扩展模型（CNN + LSTM）。

* 1. 步态识别

步态识别，首先对视频预处理将行人与背景分离，形成黑白轮廓图silhouette。然后再在连续多帧的silhouette图中获取特征，最终达到身份识别的目的，依据是每个人都会有自己的走路方式，这是一种比较复杂的行为特征。

silhouette图如下（来自CASIA-B）：



* 1. 动作识别实现

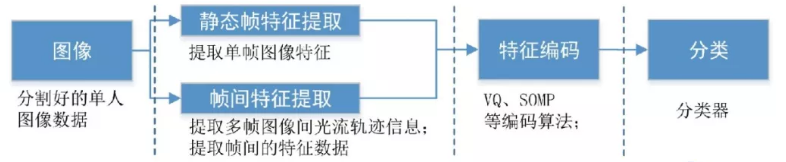
要识别出人物的动作通常需要连续的视频数据进行分析处理，需要采集的特征通常有单帧图像数据的特征和多帧图像数据之间时间上的特征，简单来说就是静态帧数据+帧间数据。

a、静态帧数据，表示单帧图像空间信息

b、帧间数据，表示时间信息（光流数据）

常用的数据集有：KTH、Weizmaan、HOHA、Keck Gesture、MSR action、YouTube Action、UT-Interaction等等。

从目标分析的角度上看，动作识别也能看作是分类的问题，每个动作就是一类，如果有二十个动作就是二十类。那么动作的识别最终就能转变成特征提取、分类两个主要过程。特征提取包含静态帧的特征，动态帧间的特征。算法在不断的迭代发展，产生了很多经典的处理模型、算法。一般的识别流程为：



从上图可以看到，动作识别的关键步骤是人体分割、特征提取以及特征编码。

人体分割是将图像中人体所在区域与图像的其他部分分割出来，使用人体关键点检测来分割的方法前文已经有所介绍；特征编码的作用主要是将提取到的特征标准化到统一的空间，而通常特征提取和特征编码一般是配套使用的，动作识别的CNN模型中已经包含了特征提取和特征编码。

上文提到，动作识别需要提取静态帧特征与帧间特征，比较经典的网络模型是Two-StreamConvolutionalNetworks、TSN、TRN等。

动作分析的模型一直在进化、在向着精度更高、速度更快的方向进化，不管是two-CNN的、3DCNN的、CNN+LSTM的等等，都需要解决下面的问题：

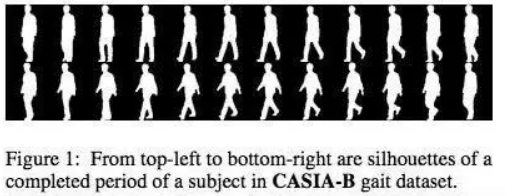
1.动作识别一般都需要一系列连续的frame,如何从video里采样到更好更合适不重复的frame；

2.一系列连续的frame往往需要消耗很大的计算力，使real time比较难；

3.动态动作幅度有大小，有快有慢，静态动作又是静止的图像，需要准确区分识别；

4.训练集的动作跟实际环境的动作会有较大差别，如何保证实际环境中的准确率是个大问题。

* 1. 步态识别实现

步态识别处理的数据不是彩色数据，而是经过处理的黑白轮廓图silhouette，silhouette图获取方式一般可以通过语义分割等方式得到。如下图：

**一般步态识别主要有两大类方法：**

**将步态看作图像**

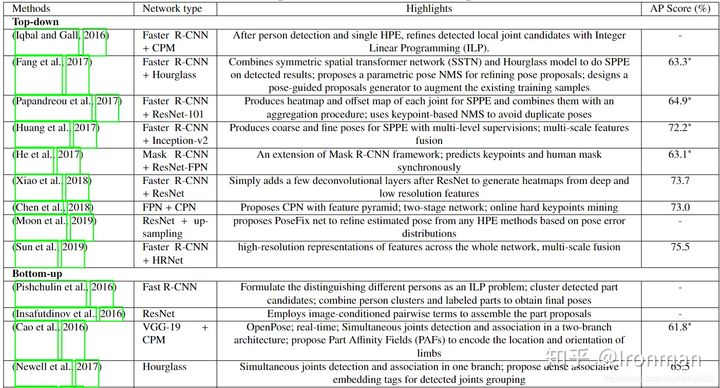
将多帧的步态轮廓图压缩成一幅图像，将步态识别看成一个图像匹配问题。优点是计算简单，缺点也同样明显，就是忽略了步态中的时序信息，也无法建模精细的空间信息，从而准确率不会高。

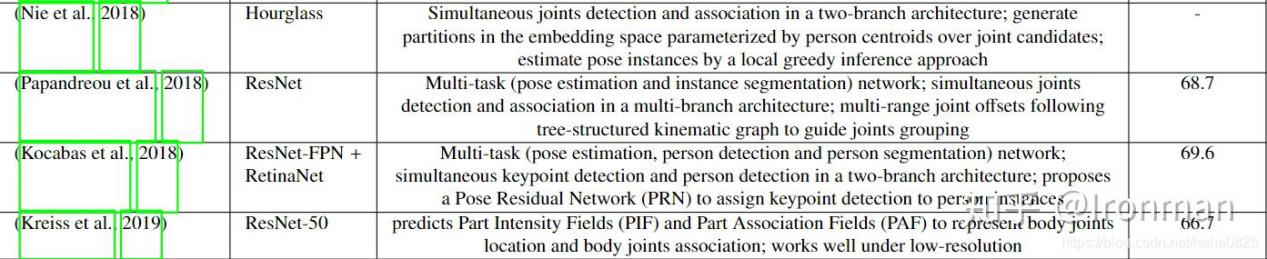
**将步态看作视频序列**

考虑直接从silhouette提取特征，分为空间（每个轮廓图像的特征）、时间（多个轮廓图像之间的信息）两个部分提取特征，优点是可以很好的建模步态中的时、空域信息，缺点是计算量大、不易训练。

###### 总结

下表是主流的多人姿态估计方法，最后一列是COCO测试开发数据集的平均精度（AP）评分。





目前，自下而上方法的处理速度非常快，有些方法可以实时运行。 但是，性能可能会受到复杂背景和人为遮挡的很大影响。 自上而下的方法在几乎所有基准数据集中都实现了最先进的性能，而处理速度受到检测到的人员数量的限制。