基于深度学习的物体检测 - 作业 7

peng00bo00

September 13, 2020

1. 本次作业是对 MTCNN 代码进行调试，现将代码核心内容总结如下：

   MTCNN 包括 PNet、RNet 和 ONet 三个串联的子网络。每个网络结构大同小异，由 backbone 和 3 个后续分支组成。backbone 包括卷积层、激活层和池化层，用来产生特征图，后续 3 个分支利用 backbone 产生的特征来预测类别、预测框位置以及关键点位置。MTCNN 预测流程如下：

   (a) 对输入图像进行缩放，使得图像中的人脸最小检测尺寸等于 PNet 的检测尺寸 (12×12)。
   (b) 将缩放后的图像输入到 PNet 中得到预测框对应的类别和位置回归值。
   (c) 根据 PNet 分类阈值（0.6）将预测框划分为正例（人脸）和反例（非人脸），得到人脸预测框。
   (d) 对预测框使用 NMS 去除重叠的部分。
   (e) 缩放当前图像并重复上述过程直至图像大小与 PNet 检测尺度相同，从而获得不同尺度的人脸预测框。
   (f) 对全部预测框使用 NMS 来去除重叠的部分。
   (g) 从原始图像中裁剪出预测框并缩放为 RNet 的检测尺寸 (48×48)。
   (h) 将缩放后的图像输入到 ONet 中得到预测框对应的类别、位置回归、以及人脸关键点位置值。
   (i) 根据 ONet 分类阈值（0.7）将预测框划分为正例（人脸）和反例（非人脸），得到人脸预测框。
   (j) 对预测框使用 NMS 去除重叠的部分。
   (k) 从原始图像中裁剪出预测框并缩放为 ONet 的检测尺寸 (24×24)。
   (l) 将缩放后的图像输入到 RNet 中得到预测框对应的类别和位置回归值。
   (m) 根据 RNet 分类阈值（0.7）将预测框划分为正例（人脸）和反例（非人脸），得到人脸预测框。
   (n) 对预测框使用 NMS 去除重叠的部分，得到最终的人脸预测框以及人脸关键点。

   使用示例图片得到人脸检测结果如 Fig.1所示。使用 CPU 进行检测耗时约为 0.67s，其中 PNet 耗时 0.45s，RNet 耗时 0.16s，ONet 耗时 0.06s；使用 GPU 进行检测耗时约为 0.68s，其中 PNet 耗时 0.47s，RNet 耗时 0.20s，ONet 耗时 0.01s。试验结果表明 MTCNN 的主要耗时在 PNet 部分，且使用 GPU 没有计算效率的提升。我认为其主要原因在于 ONet 在不断缩放原始图像进行检测，相当于使用了图像金字塔，而产生图像金字塔是在 CPU 上进行的因此使用 GPU 不会带来效率的提升。

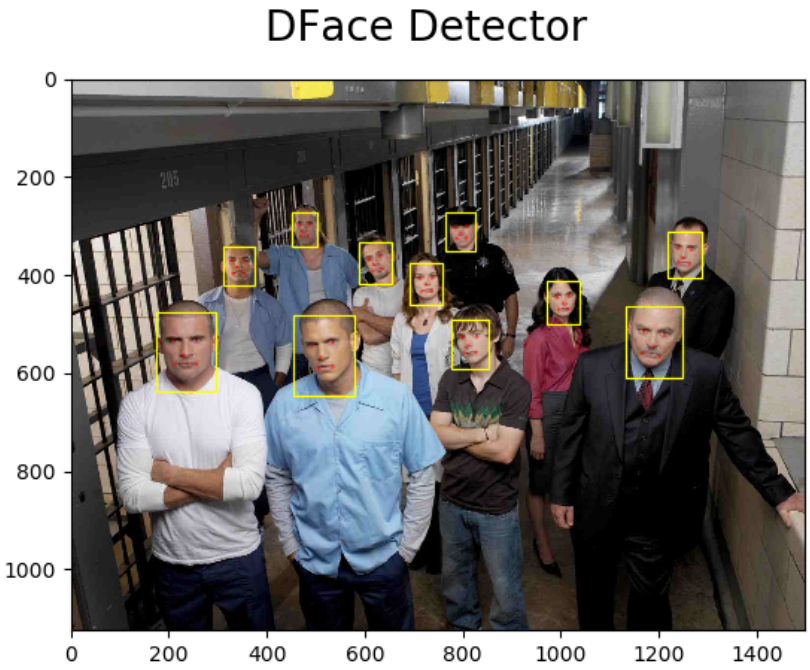

Figure 1: 人脸检测结果