

基于人脸卡口数据的行人共现关系图谱构建^①



屈诗琪, 刘宇宁, 范冰冰

(华南师范大学 计算机学院, 广州 510631)
通信作者: 范冰冰, E-mail: fanbb1962@qq.com

摘要: 随着人脸识别技术的不断进步以及人脸卡口的大范围且密集的部署, 本文针对团伙犯罪案件侦察这一应用场景, 对人脸卡口数据进行深入挖掘, 探究其中行人间的共现关系, 获取所关注的嫌疑人的现实社交网络, 锁定团伙其余人员. 经过实验比对和论证, 本文使用 Chinese Whispers 聚类算法对行人节点进行识别, 通过 Faiss 加速邻接边的构建, 加速图的初始化步骤, 解决其聚类效率低下的问题. 在此基础上, 使用共现频次和 Apriori 算法中的置信度挖掘行人间的共现关系, 构建行人共现关系图谱.

关键词: 人脸聚类; Chinese Whispers; 共现; 置信度; 图谱

引用格式: 屈诗琪, 刘宇宁, 范冰冰. 基于人脸卡口数据的行人共现关系图谱构建. 计算机系统应用, 2022, 31(6): 347-353. <http://www.c-s-a.org.cn/1003-3254/8547.html>

Construction of Co-occurrence Relationship Between Pedestrians Based on Smart Face Capture Cameras' Data

QU Shi-Qi, LIU Yu-Ning, FAN Bing-Bing

(School of Computer Science, South China Normal University, Guangzhou 510631, China)

Abstract: Nowadays, face recognition technology keeps advancing and smart face capture cameras are widely and intensively deployed. Aiming at the application scenario of gang criminal case detection, this study conducts in-depth data mining and explored the co-occurrence relationship between pedestrians. Other gang members are locked through the realistic social network of suspects. After experimental comparison and demonstration, this study uses the Chinese Whispers clustering algorithm to identify pedestrian nodes and Faiss to accelerate the construction of adjacent edges. In this way, the initialization of the graph is sped up, which improved the clustering efficiency. On this basis, the co-occurrence frequency and the confidence in the Apriori algorithm are used to mine the co-occurrence relationship between pedestrians, resulting in the graph of the co-occurrence relationship.

Key words: face clustering; Chinese Whispers; co-occurrence; confidence; graph

1 引言

随着人脸识别技术的不断突破和完善, 可以提取到人脸在不同光影、姿态、表情和成像条件下的鲁棒的特征, 进而通过人的脸部特征信息进行身份识别, 因此大范围地部署人脸卡口是现在安防领域的重要发展目标. 人脸卡口即智能人脸抓拍摄像机, 主要部署在非机动车道, 比如一些重要的道路的人行道或者一些场

所的出入口上, 只关注行人信息, 可以有效地采集经过的行人人脸图片, 形成人脸图像库. 由此, 当锁定某个案件嫌疑人时, 可以实现人脸检索功能, 得出嫌疑人被拍到的图片、地点和时间信息, 大致可以还原该人的行动轨迹; 还可以实现实时人脸比对功能, 当人脸被抓拍到时, 实时比对布控库中的人脸, 抓捕在逃人员等.

但在侦察团伙犯罪案件时, 对人脸卡口数据已经

^① 基金项目: 广东省重大科技专项 (2016B030305003)

收稿时间: 2021-09-03; 修改时间: 2021-09-29; 采用时间: 2021-10-29; csa 在线出版时间: 2022-05-26

不满足只能实现人脸检索、人脸比对和人脸验证等功能. 如何从这样一个海量的未知人员人脸图像库获得更深层的信息, 比如除了获取嫌疑人的个人信息之外, 还想知道与嫌疑人有密切来往关系的人有哪些, 即构建嫌疑人社交网络, 这将会是公共安全领域关注的热点问题. 人的社交一般分为虚拟和现实, 虚拟比如手机通话、微信等的社交软件, 这一类数据可以通过网络监管获取. 但现实的社交是比较难获取的, 现有的方法主要还是靠人工操作, 通过调取海量的监控视频进行反复查看才能建立行人之间的关系, 效率低下. 利用人脸卡口数据, 自动构建行人共现关系图谱, 能够给解决这一类问题提供思路和线索.

随着多媒体技术的快速发展, 图像、视频等多媒体数据的大量涌现, 基于图像和视频信息的社交关系理解研究也在进一步发展. Wu 等人^[1]通过人们的相册集发现人们现实的社交关系, 其中根据个体在同一张照片的次数和距离来判定关系的紧密程度建立无向加权图, 再使用一种图聚类算法, 来检测社交集群; 周丽丽^[2]将影视视频关键帧作为事务, 使用 FP-Growth 对其进行频繁项集挖掘来判定两个人物之间有无关系, 当共同出现次数超过阈值则存在关系, 以此构建人物关系网络; 周檬^[3]利用 DBSCAN 进行人脸聚类, 再通过统计监控视频关键帧中行人与行人间的共现关系、人与摄像头的出现关系构建视频图谱.

在大多社交关系的研究中, 缺乏人物节点(关系图上的顶点)识别的具体论述, 使用的聚类算法没有经过比较并对其进行有效性验证. 人脸卡口的人脸类别数量相对于相册集或者影视剧中的实验数据集更为复杂多样, 作为关系网络构建的基础, 准确地进行人物节点识别才能构建可用的行人共现关系图谱; 在目前研究人物之间的社交关系中, 其中关系存在判定, 即有无关系, 主要是基于共现的方法, 关系程度则取决于共现频度. 对此, 本文基于行人节点识别和共现关系抽取两个方面完成行人共现关系图谱的构建流程: 首先提出基于 Chinese Whispers 的行人节点识别的算法, 在此基础上使用 Faiss 加速邻接边的构建, 提高行人节点识别效率, 并与其他算法做比较验证了其优越性. 其次, 在共现频次的基础上增加其置信度来挖掘行人的关系, 最后以图形界面的形式展示图谱.

2 相关知识

2.1 人脸聚类评价方法

使用有标签的人脸数据集并根据两种聚类评价指

标对聚类结果进行评价: 1) 采用 scikit-learn 库中的聚类效果评价指标: 同质性 (homogeneity): 每个集群只包含单个类的成员; 完整性 (completeness): 同类的所有成员都分配在同一集群中; 同质性和完整性的加权平均 F -measure. 以人脸聚类为例, 同质性即同一个类中是否都是同一个人的照片, 完整性即是否将一个人的所有照片都聚集在一个类中. 2) 采用文献 [4] 中的成对准确率 $P_{pairwise}$, 成对召回率 $R_{pairwise}$ 指标. 如式 (1)、式 (2) 所示:

$$P_{pairwise} = \frac{TP}{TP + FP} \quad (1)$$

$$R_{pairwise} = \frac{TP}{TP + FN} \quad (2)$$

其中, TP 表示同一个人的人脸被正确地聚在同一类中的对数; FP 表示不是同一个人的人脸被错误地聚在同一类中的对数; TN 表示不是同一个人的人被正确地聚到不同的类中; FN 表示同一个人的人脸被错误地聚到不同的类中. 其中, 成对 F 均值度量 $F_{pairwise}$:

$$F_{pairwise} = \frac{2 \times P_{pairwise} \times R_{pairwise}}{P_{pairwise} + R_{pairwise}} \quad (3)$$

2.2 DBSCAN

DBSCAN (density-based spatial clustering of applications with noise) 是一种很典型的密度聚类算法, 这类密度聚类算法一般假定类别可以通过样本分布的紧密程度决定. 算法通过参数 (Eps , $MinPts$) 用来描述样本分布紧密程度, 其中, Eps 描述任意样本的邻域距离阈值 (距离度量一般使用欧式距离、曼哈顿距离等), $MinPts$ 描述了任意样本的 Eps 邻域中样本个数的阈值. (Eps , $MinPts$) 参数是影响聚类效果的两个很重要的参数, Eps 过小会导致原本是一个类簇却被分割成多个小的类簇, Eps 过大会导致不是一个类簇的多个距离相近的簇聚成一个大簇. $MinPts$ 越大, 形成类簇的条件就越苛刻, 那么样本集的噪声点就越多, 当 $MinPts=1$ 时, 样本集中就没有噪声点, 所有的样本都能在类簇中.

2.3 Approximate Rank Order

Approximate Rank Order^[4] 聚类是根据 Rank Order^[5] 聚类改进的. 与绝对距离 (比如欧氏距离) 不同, Rank Order 聚类发现了来自同一个人的两个脸往往有许多共享的邻居, 但是来自不同人的两个脸的邻居通常差别很大这一特点, 提出一个新的人脸相似度量方法 Rank Order 距离, 它通过比较两个样本之间的邻域结构得到样本之间的相似度. 后来, 为了使其能应用在大

规模人脸数据中,提高聚类效率, Approximate Rank Order 算法重新定义距离的度量方式,不需要计算每个人脸的所有最近邻而是 k 个最近邻,从而得出人脸之间的非对称秩序距离之后,就可以逐步地将低于距离阈值 $threshold$ 的人脸合并为一个类簇. k 和 $threshold$ 是该算法的两个需要指定的参数. k 是选取每个人脸最近邻的数量,而且 Approximate Rank Order 使用 FLANN 库的随机 k -d tree 算法求出每个人脸图片的 k 个最近邻图片排序,聚类的效率显著地提高.

3 行人共现关系图谱构建

人脸卡口主要采集经过的行人的人脸图像、被抓拍到的时间戳以及卡口自身的编号,其中卡口的编号就代表行人被抓拍时的位置信息.基于人脸卡口的行人共现关系图谱构建的整体流程如图1所示,其中最主要的两个步骤:一是基于聚类的行人节点识别,由于人脸卡口中出现的人是未知的,没有模板人脸库让其去一一对应来标明行人的身份,所以使用基于聚类的方法标注人脸卡口中的人脸图像,从而得到图谱中的行人节点;二是行人共现关系抽取,通过人脸卡口数据的特性,定义共现关系的有无和共现关系的强度来共同抽取行人间的共现关系,从而得到图谱中节点之间的边.

3.1 基于聚类的行人节点识别

基于聚类的行人节点识别,也可称作人脸聚类,旨在把相似的人脸图像归为一类,即把属于同一个人的脸打上同一个标签^[6],如图2所示.人脸聚类效果的好坏主要取决于人脸特征的提取(人脸的向量表示)和聚类算法的选择^[7],人脸特征的提取要保证具有高精度的可分性,这里使用目前效果较好的 CNN+ArcFace Loss 方法提取人脸特征向量.适用于人脸聚类的聚类算法是不需要预先设定类别的数量的,如传统的 K-means

就不行,因为我们对人脸卡口数据集是没有预先的认识,其具体包含多少个类别是未知的.文献[3]使用 CNN+ArcFace Loss 方法提取人脸特征,直接使用 DBSCAN 作为人脸聚类算法,标识监控视频中出现的行人;文献[8]使用 FaceNet 提取人脸特征,在 LFW 数据集上验证了 DBSCAN 相较于其他聚类方法(如谱聚类和 Rank Order);文献[9]使用附加间隔 Softmax 提取人脸特征,分别在 LFW 和 LFW+模糊视频上实验结果都证明 Approximate Rank Order 比 DBSCAN 更加适合处理较大规模的复杂人脸数据;文献[10]使用 Chinese Whispers 对人脸聚类获取同一个人的脸图像,结合人脸质量评估算法从而达到在一段视频中人脸图像去重的目的.可见,适用于人脸聚类的聚类算法是需要根据人脸特征提取算法和人脸数据择优选择的,常用来做人脸聚类的算法主要是 DBSCAN, Approximate Rank Order 和 Chinese Whispers.

3.1.1 Chinese Whispers 算法

Chinese Whispers^[11]最早是被用在自然语言处理如文本聚类中,后来也广泛应用于人脸聚类,它不需要预先设定类别数量,是一种可以自动查找类别个数的高效图聚类算法.该算法分为两个步骤:第一步是构建无向加权图 $G=(V, E)$.把数据集中的每一个样本当作图上的一个点 $v_i \in V$,并且初始化每一个节点的类别属性为当前节点顺序编号.接着就是构造无向图的边,当节点 v_i 和 v_j (其中 $v_i < v_j$) 的相似度大于 $threshold$ 时,就在 v_i 和 v_j 之间构造一条边 $(v_i, v_j, w_{ij}) \in E$,其中 w_{ij} 是 v_i 和 v_j 的相似度(采用如余弦相似度);第二步是迭代.随机选取一个节点 v_i ,选取该节点的邻居节点中权重最大的类别作为当前节点的类别,若邻居节点有多个属于同一类别,则将这些节点的权重相加.如此,直到所有节点循环一遍算作一次迭代.算法直到迭代次数满足 $iterations$,算法结束.

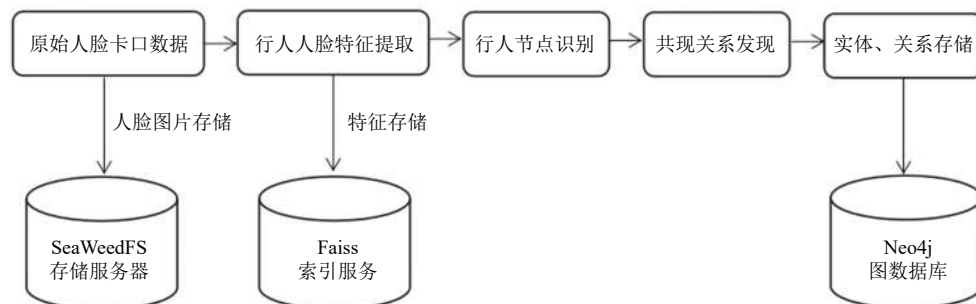


图1 行人共现关系图谱构建流程

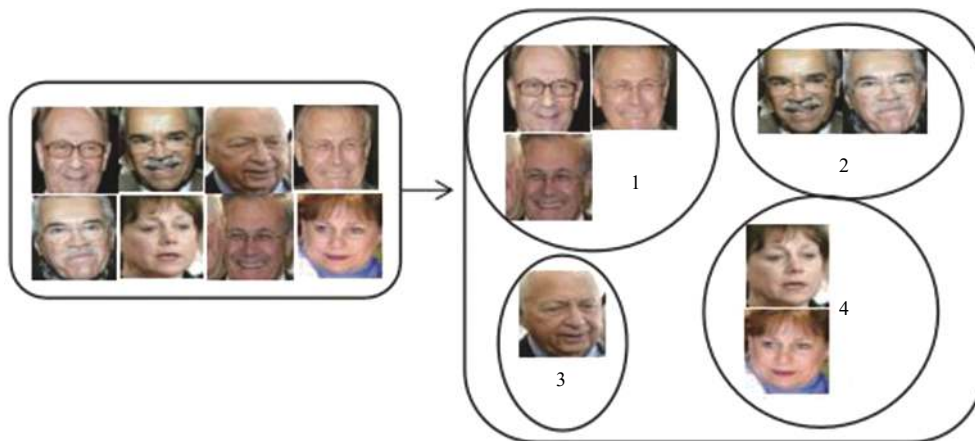


图2 行人节点识别

该算法需要事先定义两个超参数 $threshold$ 和 $iterations$. $threshold$ 是节点之间的相似度阈值, 只有大于这个阈值, 节点与节点之间才有边, 消息才得以互相传递. 形式上, 该算法并不会收敛, 所以要指定 $iterations$ 迭代次数, 但是对于加权图, 只需要几次迭代就能达到稳定状态. 它的不足之处在于因为其随机性 (对小图来说从哪个节点开始迭代很重要), 所以对于处理小图具有不确定性, 但是在大图这种不确定性将会消失, 适合处理稍大数据的聚类.

3.1.2 改进的基于 Chinese Whispers 的行人节点识别

虽然原始的 Chinese Whispers 算法是适合处理稍大数据的聚类, 但是由于人脸特征向量 512 维是高维向量, 算法需要计算不同节点之间的相似度, 其时间复杂度为 $O(n^2)$, 当处理大规模人脸聚类时, 会导致算法的第一步即图的初始化缓慢. 文献 [7] 使用增量聚类的思想提出基于 Chinese Whispers 的人脸动态聚类, 其时间效率虽然稍有提高, 但是准确率都稍有下降, 并且对于类别数较多的数据集, 时间提升效果不明显. 本文使用 Faiss (Facebook AI similarity search)^[12,13] 加速图中节点的边的构建, 把判断两节点是否相似问题, 转化为对节点的限定半径邻居查询问题, 提高聚类速度. Faiss 为稠密向量提供高效相似度搜索, 是目前最为成熟的近似邻搜索库. Faiss 提供了许多不同应用场景下的索引供选择, 由于 Chinese Whispers 算法需要计算每个样本节点之间的相似度, 即余弦相似度, 提取的人脸特征是经过 L2 范数归一化的, 人脸特征余弦相似度的值等同于人脸特征内积的值, 所以选择 Faiss 中的 IndexFlatIP 为人脸特征构建索引. 其中详细的行人节点识别步骤如算法 1 所示.

算法 1. 改进的基于 Chinese Whispers 的行人节点识别算法

输入: 行人人脸图像数据集 D , 相似度阈值 t , 迭代次数 $iterations$
输出: 行人身份标签列表 $labels$

- 1) 使用 CNN+ArcFace Loss 提取数据集 D 中所有人脸图像的人脸特征, 保存到 $feats$;
- 2) 根据 $feats$ 构建 Faiss 中的 IndexFlatIP 特征索引;
- 3) 使用特征索引, 一次性查询所有人脸特征 $feats$ 的限定半径 t 邻域的最近邻居, 保存到 res ;
- 4) 构造节点列表 $nodes$, 初始化每个节点的 $label$ 为 0 到 $len(feats)-1$;
- 5) 根据 res 构造边列表 $edges$, 其中边的权重为两个节点的余弦相似度;
- 6) 根据 $nodes$ 和 $edges$ 完成对无向图 G 的初始化;
- 7) $n=0$;
- 8) while $n < iterations$ do:
 - 打乱 $nodes$ 列表顺序;
 - for node in $nodes$:
 - 随机选取一个节点 $node$, 列出它的所有邻居节点 $nbrs$;
 - 当邻居节点中有相同 $label$ 时, 则把它们的权重相加;
 - 最后, 修改当前节点 $node$ 的 $label$ 为权重值最大的 $label$;
 - $n++$;
- 9) 迭代完成后, 得到所有节点 $nodes$ 的 $label$ 值列表, 并转换为 0 到 $(num-1)$ 值之内的标签列表 $labels$, 其中 num 是聚类后的人数.

3.1.3 实验

(1) 实验环境与数据集: 实验在 CentOS 7 系统, CPU 为 Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20 GHz, 内存为 128 GB 的服务器上进行. 使用 LFW^[14] 和 CASIA-WebFace^[15] 数据集的部分数据. 其中 LFW 数据集分布很不均匀, 大多数人只有 1 张图像, 所以只选取 LFW 中有 2 张及以上的数据集. CASIA-WebFace 的图片比较多, 由于实验室设备的限制, 只选取了一部分数据集, 如表 1.

(2) 人脸特征提取: 使用 RetinaFace^[16] 人脸检测算法对两个数据集进行数据清洗, 删除无法检测到人脸

的图片.接着使用 RetinaFace 进行人脸检测,由于有些图片中可能不止一人出现,对这些图片仅选择中心坐标的人脸作为目标对应其标签.为了更好地提取人脸特征,对 RetinaFace 人脸检测算法得到的人脸 5 个关键点进行仿射变换摆正人脸位置,再使用 CNN+ArcFace Loss 方法^[17]提取 512 维人脸特征.

表 1 实验数据集

数据集	样本数(个)	身分数(个)	人均图片数(个)
LFW	9164	1680	5
CASIA-WebFace	100000	1727	57

(3) 实验结果与分析:在人脸聚类中,我们往往会更加在意聚类评价指标中的同质性和准确性,因为如果某一个人脸的聚簇中出现大量别人的人脸,人脸聚类结果的应用就会失去它的意义,所以本文实验结果中的 F -measure 和 $F_{pairwise}$ 都是在同质性和准确率大的情况下取它们的最大值.

如表 2 所示,在 LFW 数据集的表现上,4 个算法聚类效果差距不大,但是 Chinese Whispers 的得分最高,并且簇数是最接近真值的,且经过改进的 Chinese Whispers 与原始 Chinese Whispers 算法的结果是一样的,时间效率却提高了不少.

表 2 在 LFW 子集(真实簇数=1680)上的实验结果对比

聚类算法	F -measure	$F_{pairwise}$	簇数	时间(s)
DBSCAN	0.998	0.997	1742	0.98
Chinese Whispers	0.999	0.999	1727	125
Approximate Rank Order	0.96	0.91	2519	10.8
Ours	0.999	0.999	1727	8

但是在 CASIA-WebFace 数据集的表现上差距就显现出来了,CASIA-WebFace 数据集中的图片相较于 LFW 数据集,同一个人的照片时间跨度更大、造型差异大、模糊度高等,人脸聚类的难度比 LFW 数据集大很多.如表 3 所示,Chinese Whispers 算法的表现最好,两个平均度量指标得分最高,且聚类簇数最接近真实簇数,说明其人脸聚类的鲁棒性,更适应稍复杂的人脸数据,而且经过改进后的 Chinese Whispers 使得其可以应用在大规模的数据集上.继续对我们的方法的实验结果进行详细分析,7257 个聚簇中有 5015 个孤立点,这 5012 个孤立节点是在图上没有邻接边的点,即跟样本集中的所有样本的相似度都小于相似度阈值.这些孤立节点大多是数据集采集时的错误标记或是遮挡、模糊等原因无法正确人脸识别.去掉孤立节点之后,完整

性和成对召回率提高, F -measure 得分 0.98, $F_{pairwise}$ 得分 0.96.所以,将其应用在人脸卡口数据上的聚类效果应该会更好.

表 3 在 CAISA-WebFace 子集(真实簇数=1727)上的实验结果对比

聚类算法	F -measure	$F_{pairwise}$	簇数	时间(s)
DBSCAN	0.94	0.86	16657	126
Chinese Whispers	0.97	0.91	7256	20800
Approximate Rank Order	0.89	0.67	29398	498
Ours	0.97	0.92	7257	168

3.2 行人共现关系抽取

行人共现关系抽取算法基于前面已有的基于 Chinese Whispers 的行人节点识别方法,对每张脸进行聚类,将属于同一人的人脸分配同样的编号,即 id_label ,作为不同行人的身份标识.当行人节点识别完成后,本文涉及到的数据存储字段如表 4 所示. id 表示抓拍到的人脸编号, id_label 是聚类之后的编号,相同身份的行人的 id_label 与 id 是一对多的关系.人脸图片使用 SeaweedFS^[18] 存储,使用 fid 存储图片链接,极大地减少了存储空间,且直接通过 $http$ 协议访问该链接,能够提高图片的访问速度.

表 4 存储字段描述

名称	类型	说明
id	int	人脸卡口抓拍记录ID
fid	varchar	行人人脸图片存储位置
$kakou_id$	int	卡口编号
$timestamp$	timestamp	行人被抓拍到的时间戳
id_label	varchar	行人人脸聚类分配的ID

在文献计量研究领域中,共同出现的特征项之间存在着某种关联,关联程度可以使用共现频次来评估.所以,在图像或视频的社交关系挖掘中,也是在共现的基础上发现社交关系的,大都是统计图像或视频关键帧中的共现次数.人脸卡口数据与图像和视频数据不同,共现不是体现在一同出现在一个画面中,而是体现在一定时间间隔内被同一个人脸摄像机抓拍到.当基于人脸卡口的行人共现关系网络 $G=(V, E)$,关系如公式(4)所示:

$$E_{ij} = \begin{cases} 1, & \text{if } support(i, j) \geq sup_min \\ 0, & \text{else} \end{cases} \quad (4)$$

其中, $support(i, j)$ 是行人 i, j 共同出现的次数, sup_min 是设置的共现关系的最低阈值, $E_{ij}=1$ 表示行人 i, j 之

间有边连接,即认为*i, j*之间有关系.反之,则认为*i, j*之间无边连接,即*i, j*之间无关系.一般情况下,我们把只共现一次视为偶然情况,所以 *sup_min* 要大于或等于 2.

行人对之间的关联程度除了用共现频次来表示以外,本文引入 Apriori 关联规则算法中置信度概念^[19]来计算共现关系的置信度,如式(5)所示:

$$confidence(i \Rightarrow j) = \frac{support(i, j)}{support(i)} = p(j|i) \quad (5)$$

其中, *confidence(i ⇒ j)*得到的是*i*出现,则*j*也一同出现的概率,反映了在*i*方面,认为*j*的重要程度.我们还可以得到其平均置信度,如式(6)所示:

$$confidence(i, j) = \frac{confidence(i \Rightarrow j) + confidence(j \Rightarrow i)}{2} \quad (6)$$

其中, *confidence(j ⇒ i)*是式(5)中*i, j*互换位置的结果.使用置信度来体现行人对之间的关联紧密程度更具有说服力,比如当行人*a, b*分别被拍到了2次,且他们也正好共现了2次,那么他们共现的置信度就是1;而行人*c, d*分别被拍到了10次和15次,共现了3次,那么他们共现的置信度是0.25.如果仅根据共现频次,*c*和*d*之间的关联程度要大于*a*和*b*,但是从现实角度,我们会认为*a*和*b*的关联程度要大于*c*和*d*.所以,人脸卡口中行人对满足最小支持度和最小置信度的共现关系,被认为是强关联关系,其中共现关系抽取的详细步骤如算法2所示.

算法2. 行人共现关系抽取算法

输入: 人脸卡口数据集 *D*, 卡口列表 *L*, 时间间隔 *t*, 最小支持度 *sup_min*, 最小置信度 *conf_min*
输出: 共现关系结果集合 *R*

- 1) 扫描人脸卡口数据集,统计每一个 *ID* 出现的总次数,保存到字典 *id_to_counts*;
- 2) for *L_i* in *L*:
- 3) *S* 为在 *L_i* 被抓拍的人脸集
- 4) 对 *S* 中出现的人脸按照时间戳从小到大排序
- 5) for *S_k* in *S*[0:]:
- 6) *S_j* in [*k*+1:]:
- 7) if (*T_{S_j}* - *T_{S_k}* > *t*): break;
- 8) //在时间间隔内
- 9) if (*ID_k* > *ID_j*): 交换 *ID_k* 和 *ID_j*
- 10) *R*[(*ID_k*, *ID_j*)] += 1 //记录一次共现
- 11) 把 *R* 中记录的共现次数小于 *sup_min* 的行人对过滤掉;
- 12) for *R_i* in *R*:

- 13) 根据 *id_to_count* 分别计算 *R_i* 中行人对的置信度 *conf1, conf2*;
- 14) if (avg(*conf1, conf2*) < *conf_min*): 过滤掉;
- 15) 输出 *R*

3.3 实体、关系存储

选择 Neo4j 作为图谱的图数据库存储, Neo4j 能够高效地存储节点、关系和属性,同时, Neo4j 支持 Cypher 查询语言,关系查询灵活高效,支持增删改查功能.基于人脸卡口数据的行人共现关系图谱的实体和关系存储的属性如表5所示.

表5 Neo4j 存储描述

实体	关系
实体标签: Person	关系标签: Co-occurrence
属性1: id_label	属性1: conf
属性2: img_url	属性2: times
属性3: paths	

表5中,行人节点的 *id_label* 作为唯一性约束, *img_url* 是该 ID 行人人脸图像, *paths* 是一个列表,存储了该行人的 (*kakou_id, timestamp*), 即行动轨迹.关系属性中的 *conf* 是共现置信度, *times* 是共现频数.

4 图谱可视化

根据公开的人脸数据集 LFW 和 CASIA-WebFace 随机+人为制造了人脸卡口数据集作为样例,构建行人共现图谱.最后,采用 Web 页面作为前端展示页面,使用 D3.js 作为前端展示框架,借助 D3.js 的力学导向图,将数据以网络的形式展现.如图3所示,展示了数据集中部分数据,生动地反映行人之间的共现关系,其中节点之间边的长度取决于共现关系的置信度的大小,置信度越高,节点相距就越近,更能显示紧密的关系.

5 结论

本文针对侦察团伙犯罪案件时人脸卡口数据应用的局限性以及对人的现实社交关系难以辨认的问题,实现了一种基于人脸卡口数据的行人共现关系图谱的构建流程.该流程首先使用 Faiss 加速 Chinese Whispers 算法中初始图的邻接边的构建,使其能够完成大规模人脸聚类的任务,得到人脸卡口数据这一未知人员库中行人的潜在身份 *id*,然后再根据人脸卡口记录挖掘行人的共现关系,通过共现频次和置信度来体现关系强度,最后使用前端框架展示图谱交互式界面.

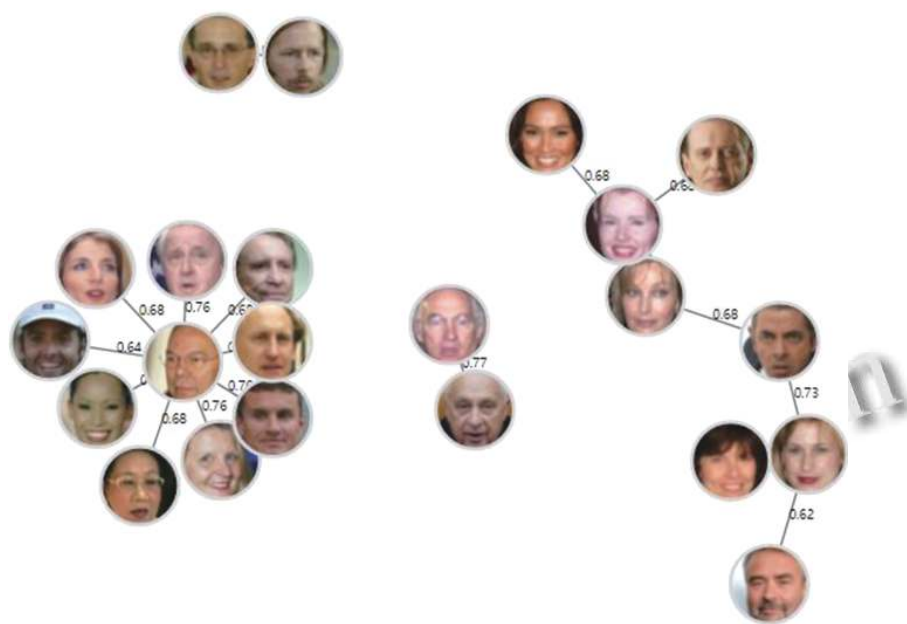


图3 基于人脸卡口数据的行人共现关系图谱展示界面

参考文献

- Wu P, Tretter D. Close & closer: Social cluster and closeness from photo collections. Proceedings of the 17th ACM International Conference on Multimedia. New York: Association for Computing Machinery, 2009. 709–712. [doi: [10.1145/1631272.1631394](https://doi.org/10.1145/1631272.1631394)]
- 周丽丽. 视频人物社交关系网络构建方的研究与实现 [硕士学位论文]. 北京: 北京邮电大学, 2019.
- 周檬. 基于监控视频的视频图谱构建和挖掘研究 [硕士学位论文]. 武汉: 华中科技大学, 2019.
- Otto C, Wang DY, Jain AK. Clustering millions of faces by identity. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(2): 289–303. [doi: [10.1109/TPAMI.2017.2679100](https://doi.org/10.1109/TPAMI.2017.2679100)]
- Zhu CH, Wen F, Sun J. A rank-order distance based clustering algorithm for face tagging. CVPR 2011. Colorado Springs: IEEE, 2011. 481–488. [doi: [10.1109/CVPR.2011.5995680](https://doi.org/10.1109/CVPR.2011.5995680)]
- 毛晓东, 董辉. 人脸聚类技术在公安行业应用浅析. 中国安全防范技术与应用, 2019, (1): 21–22. [doi: [10.3969/j.issn.1672-1470.2019.01.006](https://doi.org/10.3969/j.issn.1672-1470.2019.01.006)]
- 张陈欢, 史燕中. 基于 Chinese Whispers 的人脸动态聚类. 计算机技术与发展, 2019, 29(11): 92–96. [doi: [10.3969/j.issn.1673-629X.2019.11.019](https://doi.org/10.3969/j.issn.1673-629X.2019.11.019)]
- Farukh RS. 基于 DBSCAN 的人脸聚类 [硕士学位论文]. 广州: 华南理工大学, 2019.
- 王锟朋, 高兴宇. 基于附加间隔 Softmax 特征的人脸聚类算法. 计算机应用与软件, 2020, 37(2): 111–117. [doi: [10.3969/j.issn.1000-386x.2020.02.017](https://doi.org/10.3969/j.issn.1000-386x.2020.02.017)]
- 林增敏, 洪朝群, 庄蔚蔚. 融合人脸跟踪和聚类的人脸图像去重方法. 计算机科学, 2020, 47(S2): 615–619.
- Biemann C. Chinese whispers: An efficient graph clustering algorithm and its application to natural language processing problems. Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing. Association for Computational Linguistics, 2006. 73–80.
- GitHub. A library for efficient similarity search and clustering of dense vectors. <https://github.com/facebookresearch/faiss>.
- Johnson J, Douze M, Jégou H. Billion-scale similarity search with GPUs. IEEE Transactions on Big Data, 2021, 7(3): 535–547. [doi: [10.1109/TBDATA.2019.2921572](https://doi.org/10.1109/TBDATA.2019.2921572)]
- Huang G B, Mattar M, Berg T, et al. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Marseille, France, 2008.
- Yi D, Lei Z, Liao SC, et al. Learning face representation from scratch. arXiv: 1411.7923, 2014.
- Deng JK, Guo J, Zhou YX, et al. RetinaFace: Single-stage dense face localisation in the wild. arXiv: 1905.00641, 2019.
- Deng JK, Guo J, Xue NN, et al. ArcFace: Additive angular margin loss for deep face recognition. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach: IEEE, 2019. 4685–4694. [doi: [10.1109/CVPR.2019.00482](https://doi.org/10.1109/CVPR.2019.00482)]
- GitHub. SeaweedFS Introduction. <https://github.com/chrislusf/seaweedfs#introduction>.
- 任鸽, 吴猛, 汗古丽·力提甫, 等. 基于改进 Apriori 算法的高校课程预警规则库构建. 计算机系统应用, 2021, 30(7): 290–295. [doi: [10.15888/j.cnki.csa.008040](https://doi.org/10.15888/j.cnki.csa.008040)]