

基于 HOG 和 Haar 特征的行人追踪算法研究

陆星家 陈志荣 尹天鹤 杨帆

(宁波工程学院理学院 宁波 315211)

摘要 行人在真实场景的检测和追踪是多目标检测和追踪研究中的一个重要问题,尤其是在真实的三维场景中的多行人之间的遮挡、拥挤以及背景的变化对多目标检测和追踪研究造成了严重的挑战。在多目标检测中利用了 Haar 特征、HOG 特征,在行人正面向相机运动时,采用 Haar 特征检测器检测人脸,并结合 Haar 运动模型完成行人的检测,当行人侧向运动时,采用 HOG 特征,利用层次-部分模型进行行人的检测和追踪,在完成行人的检测之后,利用最大权重独立集算法完成帧间目标的关联。通过对 ETH、TUD 以及本地样本库的检测和追踪结果表明,采用 Haar 特征和 HOG 特征的检测算法对于行人的正面和侧面都具有较高的检测准确率、精确度。

关键词 Haar-Like 特征, HOG 特征, 层次-部分模型, Haar 运动模型, 最大权重独立集

中图分类号 TP391 **文献标识码** A

Research of Pedestrian Tracking Based on HOG Feature and Haar Feature

LU Xing-jia CHEN Zhi-rong YIN Tian-he YANG Fan

(College of Science, Ningbo University of Technology, Ningbo 315211, China)

Abstract The detection and tracking of pedestrians in the real scene is one of the most concerned questions in the multi-people detection and tracking, especially under the condition that the occlusion, congestion and varied backgrounds of people have greatly challenged the research of multi-people detection and tracking. Haar and HOG features are applied in this research. The Haar-Like feature detector is used to detect the face when people come face to face with the camera, while the Part-Based Template is used to detect and track people in people's sideway movement. After that, the maximum weight independent set is used to accomplish the relationship between the frames. The result of the detection and tracking from the ETH, TUD and local sample library attests that the hybrid algorithm has much higher accuracy and precision in detecting pedestrian's front and side face.

Keywords Haar-like feature, HOG feature, Part-based model, Haar motion model, Maximum weight independent set

近年来,随着计算机视觉技术的快速发展,目标追踪研究逐渐由单目标追踪转为多目标追踪。多目标追踪在军事和民用领域都有广泛的应用^[1],在军事方面,如无人机的目标追踪;在民用领域,如智慧交通中的车辆追踪,葛军峰在自动驾驶领域利用近红外特征和 Haar 以及 HOG 特征对行人进行追踪研究^[12],该方案采用近红外特征,使得动态的行人目标追踪率显著提高,但是由于采用多传感器,设备价格过于昂贵,该方案的推广受到制约。在智慧城市中的行人检测中,Geismann 利用二阶段的 Haar 特征和 HOG 特征对行人进行追踪研究^[13],Geismann 通过检测器实时检测分类追踪目标,该研究在图像背景较为复杂、清晰度高的视频中会造成追踪计算量的急剧攀升,准确性和可靠性受到明显的制约。采用单一相机获取可见光谱图像或视频数据,并通过检测和追踪算法进行多目标追踪成为多目标追踪研究中的发展趋势。

传统的多目标追踪算法普遍利用相似度函数或距离函数对不同帧的检测目标进行关联,选择关联度最大的帧间目标进行匹配。在 JPDA (Joint Probabilistic Data-Association) 算法中, Cox 利用追踪切分、联合似然、多假设算法完成多目标

追踪^[2],由于目标关联算法是通过概率的形式表示不同目标之间的关联关系,为了减少由于样本方差对数据的关联的影响, Cox 利用马氏距离 (Mahalanobis Distance) 对不同样本方差进行归一化,减少了样本方差对数据关联的影响,改善了多目标追踪的效果。Kenji Okuma 利用 Boost Particle Filter 算法对冰球比赛队员进行多目标追踪,取得了较好的效果^[3];通过颜色直方图提取不同目标的特征,利用混合高斯模型和 AdaBoost 算法提升了多目标追踪的准确率。Brendel 在 2011 年提出利用最大权重独立集算法对多目标进行追踪^[4],最大权重独立集算法相比粒子滤波器等算法具有简单、有效的特点,同时可以结合目标检测分类器,利用目标检测器完成目标的检测,并在检测的基础上进行追踪。

本文利用层次-部分模型进行行人检测中,结合行人运动模型检测运动的行人,在行人运动模型检测中,为了提高多行人追踪的准确率,利用最大权重独立集算法对检测到的目标进行帧间关联。最大权重独立集算法由于需要追踪的目标不能脱离检测视野,同时对于目标的遮挡等情况下,追踪效果也不理想,因此通过重检测的方法,当检测目标出现遮挡、脱离

本文受国家自然科学基金项目(40901241),浙江省自然科学基金项目(Y5090377),浙江省教育厅基金项目(Y201225208),宁波市自然科学基金项目(2012A610020)资助。

陆星家(1979-),男,博士,讲师,主要研究方向为计算机视觉、机器学习, E-mail: shlxj800@gmail.com.

检测视野等情况时,重新对视野进行检测。

1 行人检测与追踪

在行人追踪中,复杂的追踪场景以及长时间的遮挡是行人追踪中的主要问题,根据遮挡的时间又可以分为长时遮挡和短时遮挡;而根据遮挡的类型可以分为部分遮挡和完全遮挡。传统的行人目标追踪算法对以上两种类型的遮挡问题解决较差,追踪的准确率不高,德国达姆施塔特工业大学的 Andriluka 采用检测-追踪-检测的框架^[5],结合肢体为基础的结构模型,利用分层高斯过程隐变量模型,在多行人追踪和长时间遮挡中取得了较为良好的效果,是目前多行人和长时间遮挡环境下行人追踪的主流方法(见图 1),该方法可以有效地提高目标追踪的效果。在行人追踪中,Leibe 提出的密集行人检测和轨迹追踪算法是目前追踪较好的多行人追踪算法^[6]。在行人追踪前,首先需要对行人进行检测,广泛采用的是 Haar 特征、HOG 特征、SIFT 特征。通过以上特征检测行人,即将行人特征与分类器进行比较,从而检测行人。在完成行人检测之后,需要对行人进行追踪,当在追踪过程中出现长时间的遮挡、消失等情况时为了提高追踪的效果,需要对目标进行重新检测,如果没有遮挡的发生,利用数据关联算法进行目标的关联。



图 1 行人检测和追踪

在目标追踪过程中,由于行人是非刚体物体,在行人运动时外观变化较大,传统地采用粒子滤波器的多行人追踪算法对行人进行追踪,对于行人在运动过程中外观的变化以及行人之间的遮挡还不能够完全适应。在国内,中科大的程有龙等通过离线训练带有行人的先验知识的分类器,利用 Online Boosting 算法对行人进行检测和追踪^[7]。李娟等利用 Kalman 滤波器对行人目标进行追踪研究,但行人目标在运动过程会出现遮挡、消失等情况,同时 Kalman 滤波对线性模型可以较好地进行估计,而对非线性、非高斯模型的估计的准确度较差^[8]。对于多目标追踪问题,由于不同目标之间具有一定的关联性,因此,如何确定目标的关联是多目标追踪最先要解决的难题。

1.1 基于层次-部分模型的行人检测

本文在行人检测中主要利用 Pedro F. Felzenszwalb 提出的层次部分模型^[9]。层次部分模型利用 HOG 特征来获取行人的特征模板,与传统的模板不同,层次-部分模型是一个模板集合,包括根模板、部分模板。对于行人而言,利用 HOG 特征有利于从复杂的背景中提取行人的轮廓模板。相比较传统 Haar 特征,HOG 特征由于对纹理有较好的检测效果,被广泛地用于图像处理和目标检测中。由于 HOG 算法提取的特征具有很好的通用性,为了提高 HOG 算法检测目标的速度,将检测图像进行金字塔算法,可以有效地提高算法的检测速度(见图 2)。



图 2 基于层次-部分模型的行人检测器

1.2 基于 Haar 特征的运动检测

在完成行人的检测之后,需要对检测的行人进行追踪,在追踪过程中,常见的追踪算法包括 Particle Filter, Boosting 算法等。本文采用 Haar 特征的运动检测器对行人进行追踪, Haar 运动模型由 Paul 提出^[10],通过 Haar 特征对行人进行追踪,Paul 采用的 Haar 运动特征如图 3 所示,运动检测算法借鉴人脸检测中的 Cascade-AdaBoost 算法,通过将决策树叠加,利用 Boost 提升方法获得强分类器。Cascade-Boost 算法需要利用 Haar 小波提取检测图像的特征,AdaBoost 算法通过构建一组弱分类器,并将弱分类器加权叠加,构建强分类器,AdaBoost 算法通过训练样本获得检测器,算法初始阶段采用均匀分布。将每个样本的权重初始化为 $1/m$,由于没有样本数据的先验信息,因此采用均匀分布作为各个样本的权重较为合适。同时设置相关的分类器 $h_t(i)$,设置算法的运行时间为 $t=1, \dots, T$ 。

$$h_t(i) = \arg \min_{h_j \in H} \xi_j \quad (1)$$

$$\xi_j = \alpha_j = \frac{1}{2} \ln \left(\frac{1 - \epsilon_j}{\epsilon_j} \right) \quad (2)$$

更新每个样本的权重,

$$\omega_{t+1,i} = \frac{\omega_{t,i}}{\sum_{i=1}^m \omega_{t,i}} \exp(-\alpha_t y_t(i) h_t(i)) \quad (3)$$

对于两类问题而言,标签数据可以取 $\{-1, +1\}$ 。



图 3 Haar-Like 运动特征

在目标检测过程中,正确分类的权重缩小,错误分类的权重增大。利用基于 Haar 特征的运动模型对运动的行人进行追踪,分辨率较高的视频以及运动的行人,可以利用 Haar 运动特征进行行人追踪。

2 多行人目标追踪

在多行人目标追踪研究中,采集的图像或视频的背景、图像的清晰度、光照的变化以及严重的遮挡都是多行人目标追踪遇到的障碍。由于在不同时间段,采集的图像或视频中光照的变化也会造成背景的变化,为了提高目标追踪的准确率,通过前景提取较为简单的追踪目标背景。

2.1 数据关联算法

常见的数据关联算法包括:联合似然估计、多假设算法,其中联合似然估计算法通过对不同情况的联合估计来求追踪目标的最大联合估计,而多假设算法利用假设检验的思想对不同的追踪目标的准确率进行估计,利用概率寻找准确的命题。

为了能够及时地对检测目标进行追踪,需要对检测目标

进行帧间数据关联,常见的数据关联算法 JPDA,为了克服数据关联算法对于追踪目标在视频中的变化,通常采用马氏距离(见式(4))作为追踪目标帧间的度量函数,马氏距离可以有效地减少不同类型检测数据之间由于度量尺度对追踪目标所造成的影响。

$$d_s(x, y) = \sqrt{(x-y)^T S^{-1} (x-y)} \quad (4)$$

在马氏距离中 S^{-1} 表示上一帧检测目标的协方差逆矩阵,该矩阵是一个正定对称矩阵。

2.2 最大权重独立集算法

最大权重独立集算法是由 M. Pavan and M. Pelillo 在 2007 年提出的^[12]。William Brendel 对该算法进行了改进,并将该算法应用于多目标追踪问题研究,通过在不同帧之间进行帧间匹配来寻找最大权重独立集,每个独立集就是一个追踪目标的匹配。由于 G 的最大权重独立集等于每个独立子图 $G^{(t)}, t=1, \dots, T$ 的最大权重独立集的联合,先分别解决各个 $G^{(t)}$ 的最大权重独立集问题,用 $\Sigma^{(t)}$ 来表示。然后将短轨迹连接到不同的轨迹,这样一个轨迹 T 可能仅包含一个来自 $\Sigma^{(t)}, t=1, \dots, T$ 的短轨迹,如果 $i^{(t)}$ 和 $j^{(t+1)}$ 分别从同一对象探测为结束和开始,那么 T 可能仅包含两个连续短轨迹 $i^{(t)} \in \Sigma^{(t)}$ 和 $j^{(t+1)} \in \Sigma^{(t+1)}$ 。

$$x^* = \arg \max_x \sum_{i \in V} w_i x_i \prod_{j \in V, (i,j) \in E} (1-x_j) \quad (5)$$

满足 $\forall i \in V, x_i \in \{0, 1\}$ 。

$$y^* = \arg \max_y \sum_{i \in V} w_i \sigma(y_i) \prod_{i \in V} (1-\sigma(y_i))^{B_{ij}} \quad (6)$$

满足 $\forall i \in V, x_i \in \{0, 1\}$ 。

$$\dot{y}_i = (1-\sigma(y_i))w_i h_i - \sum_j B_{ij} \sigma(y_j) |w_j h_j| \quad (7)$$

利用式(5)来表示最大权重独立集问题的精确离散方式,如组合优化中的一般化,通过将离散等式松弛为连续定义域。定义辅助的实向量 $y=(y_i) \in R^n$,并用 S 型函数 $x_i = \sigma(y_i) = (1+e^{-\beta y_i})^{-1}$ 来替换约束 $\forall i \in V, x_i \in \{0, 1\}$,对于 S 型函数,通常将 $\beta=10$ 。得到连续方程式(6),采用最大权重独立集算法迭代寻找式(7)的最优解 $y^* \in R^n$ 。在每次迭代 τ 中,当前的解决方案会通过一阶动态 $y^{(\tau)} = y^{(\tau-1)} + \Delta\tau \dot{y}^{(\tau-1)}$ 来进行更新。最终得到式(7),它在一阶动态方程中被用于获得下一个迭代的解决方案,直到收敛为止。

表 1 最大权重独立集算法

算法 1 最大权重独立集算法
输入: 图 G
输出: G 的最大权重独立集
1. 用 $y_i^{(0)} \in (-1, 1)$ 初始化任意 $y^{(0)}$;
2. 计算
$\forall i \in V, h_i = \sigma(y_i) \prod_{j \in V} (1-\sigma(y_j))^{B_{ij}}$;
3. 用式(3)~式(7)中的方法计算 $\dot{y}^{(0)}$;
4. 当 $\ \dot{y}^{(\tau)}\ _2 > 0$ 时, 循环
5. $\Delta\tau \leftarrow \text{LineSearch}(y)$;
6. $y^{(\tau+1)} \leftarrow y^{(\tau)} + \Delta\tau \dot{y}^{(\tau)}$;
7. $\forall i \in V, h_i^{(\tau+1)} = \sigma(y_i^{(\tau+1)}) \prod_{j \in V} (1-\sigma(y_j^{(\tau+1)}))^{B_{ij}}$;
8. 用式(3)中的方法更新 $\dot{y}^{(\tau+1)}$;
9. 结束
10. $y^* = y^{(\tau+1)}$;
11. 返回 $\forall i \in V, x_i^* = \sigma(y_i^*)$

2.3 基于最大权重独立集的多人追踪

通过层次部分模型算法进行行人检测,然后利用 Haar 运动特征进行行人运动追踪,为了提高行人追踪的准确率,采用检测-追踪-再检测的框架。追踪的过程利用最大权重独立

集完成目标的关联,利用式(5)进行检测目标的关联优化,由于式(5)是一个整数规划问题,且在整数规划中, x 的取值是 $\{0, 1\}$ 的示性函数。为了能够对两个目标进行关联, z 和 z' 分别表示不同帧的检测目标。为了能够度量 z 和 z' 的距离,采用式(8)对追踪目标进行关联。

$$w = \exp(-(z-z')^T M (z-z')) \quad (8)$$

通过拉格朗日松弛算法将整数规划问题转化为连续区域的线性规划问题,通过求解线性规划的最优解寻找检测器之间的最大匹配权重,最大匹配的权重构成短时间的行人追踪器。

3 检测器训练与追踪器的验证

3.1 检测器的训练

为了完成行人目标检测,首先利用样本库对行人特征进行学习,在训练过程中,将样本库分为正面行人数据集和侧面行人数据集,对正样本和负样本进行标记,为了提高检测器的训练精度,采用 Cross-Valid 方式将正负样本分为 5 组,每次选取随机选取 4 组作为训练集,然后利用剩余的一组进行验证。通过对样本数据的训练,本文获取的层次一部分的行人检测模板如图 4 所示,图 4 中第一个检测器是低分辨率的根检测器,第二个检测器是高分辨率的检测,第三个检测器是低分辨率检测器和高分辨率检测器之间的层次一部分模型。

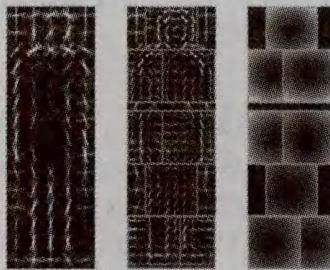


图 4 HOG 特征的层次一部分行人检测模板

3.2 检测器的检验

在完成层次一部分行人检测模板的训练之后,需要利用层次一部分行人检测模板对行人进行检测测试。图 5 是利用采集的行人数据进行测试的结果,在图 5 中第一排,对于具有较多遮挡的行人检测,漏检的行人为 3 个;而在第二排,行人的误检和错检率较高,总共 18 个行人,只检测到 9 个行人,漏检率为 50%,因此基本的层次一部分行人检测模板不能完全适应行人目标的检测和追踪。本文在基本的行人检测器基础上,增加了侧面和有倾斜角度的行人检测器。



图 5 基于层次一部分模型的行人检测器

4 仿真实验

本系统是在一定的测试环境下实现的,其中使用的硬件

是 DuoCore2. 8G 的 CPU, 在 WinXP 系统下, 利用 OpenCV 2.3 库在 Visual Studio 2005 环境下进行测试。测试中用来进行定量估计的数据集是 ETHZ Central、TUD Crossing、I_Lids 数据集以及宁波工程学院监控视频。这些数据集的录像是用静态和移动的相机进行拍摄的。目标是从变化的视角和遮挡下得到的, 目标也会有不同形态的运动。我们的数据集提出了一种宽范围的挑战: 杂乱的背景、遮挡、非静态相机, 以及规模的变化。

为了正确反映出基于最大权重独立集多目标行人追踪算法的准确率, 本文主要从 3 个指标上对该算法进行度量: 1) 正确检测率 (TPR), 即灵敏度; 2) 假阳性 (FPR); 3) 假阴性 (FNR) 指标。TPR 主要检测算法对于行人追踪的准确性, 是反映该算法追踪准确率最重要的指标; FPR 主要是度量算法的误检率; FNR 主要是度量算法的漏检率。

$$TPR = \frac{TP}{TP + FN} \times 100\% \quad (9)$$

误检率对于多目标追踪算法而言, 当算法的误检率较高时, 该算法的可用性就较差, 这是由于追踪过程中错误较多, 不能满足多目标追踪算法的要求。

$$TNR = \frac{TN}{TN + FP} \times 100\% \quad (10)$$

漏检率同样是多目标追踪算法的一个重要指标, 漏检是指在追踪过程中, 一些目标随着追踪过程被遗漏了, 该算法的漏检率较低时, 算法可用性较好。

$$FPR = \frac{FP}{FP + TN} \times 100\% \quad (11)$$

为了验证算法的精密度 (Precision) 和准确度 (Accuracy), 本文选择了几个具有一定挑战难度的数据库进行验证: ETHZ Central 数据库、TUD Crossing 数据库、i-Lids AB 数据库以及宁波工程学院 (NBUT) 的数据。

表 2 测试数据集的检测精度和准确度

数据集	精密度/%	准确度/%	漏检率/%	错检率/%
ETH	74.0	74.2	21.7	0.7
TUD	62.4	79.7	19.5	1.1
I_Lids	72.5	78.6	19.4	1.5
NBUT	82.0	82.3	18.05	0.0

4.1 行人目标的漏追踪率分析



图 6 侧面运动的行人追踪

利用本算法对 TDU 数据集的测试结果如图 6 所示。利用层次部分模板首先完成行人的检测, 然后利用 Haar 运动模板对运动的行人进行追踪, 在图 6 的第一排, 对于单一行人的检测较为准确, 始终能够对运动的行人进行追踪, 没有出现误追踪和错追踪的情况。在图 6 的第二排, 视频中同时出现 7 个行人, 由于遮挡的原因, 检测器只能检测到 4 个行人, 并对 4 个行人进行追踪。

4.2 行人目标的误追踪率分析

根据以上两个统计指标可以得出我们的平均正确检测率达到 75% 左右, 由于实际情况中的不定因素有很多, 如复杂的背景、图像的清晰度等等, 因此当待检测的图像清晰度较高时, 检测的正确率也较高; 当待测试图像中的人形区域比较模糊时, 识别率就会较低, 这与当前获取的人形姿态以及摄像头的清晰度都有很大的关系。在测试过程中也发现检测的效果会受到光线的影响, 如对光线过亮或过暗时拍摄的人形区域检测时会发生错误, 因此图像的预处理非常重要, 可加入自动光线补偿过程使得过亮或过暗的光线得到调整, 从而不影响人形区域的定位。另外, 在测试的过程中发现对于从视频序列中如何选择拍摄角度与效果较好的一帧也是需要改进的地方, 因此前期帧质量的选择会影响到后续的特征提取与检测。正面运动的行人追踪如图 7 所示。



图 7 正面运动的行人追踪

结束语 本文介绍了一种基于最大权重独立集的数据关联算法。通过与层次部分模型、Haar 运动模型的结合, 组成基于 Harr 运动和层次部分模型的复合检测器, 针对复杂场景中的多行人追踪问题, 首先采用层次部分模型进行行人检测, 在完成行人的检测之后, 利用 Haar 运动模型进行行人的追踪, 并利用最大权重独立集算法构建帧间行人目标的关系, 对于长期遮挡的行人, 采用层次部分模型进行重新检测, 降低了行人的误追踪率和错追踪率。通过 ETHZ、TUD、I_Lids 以及宁波工程学院视频数据集的追踪结果发现, 该混合算法对于地铁行人检测、火车站行人检测以及道路行人检测均具有良好的追踪效果。

参考文献

- [1] Xiao Jiang-jian, et al. Vehicle Detection and Tracking in Wide Field-of-View Aerial Video [C]// IEEE Conf. on Computer Vision and Pattern Recognition. 2010; 13-18
- [2] Cox I J, A review of statistical data association techniques for motion correspondence [J]. International Journal of Computer Vision, 1993, 10(1): 53-66
- [3] Okuma K, Taleghani A, Freitas N D, et al. A boosted particle filter: Multitarget detection and tracking [C]// European Conference on Computer Vision. 2004; 28-39
- [4] Brendel W, Amer M, Todorovic S. Multiobject tracking as maximum weight independent set [C]// IEEE Conf. on Computer Vision and Pattern Recognition. 2011; 1273-1280
- [5] Andriluka M, Roth S, Schiele B. People tracking by detection and people detection by tracking [C]// IEEE Conf. on Computer Vision and Pattern Recognition. 2008
- [6] Leibe B, Schindler K, Gool L V. Coupled detection and trajectory estimation for multi-object tracking [C]// IEEE International

- [7] 程有龙,李斌,张文聪,等.融合先验知识的自适应行人跟踪算法[J].模式识别与人工智能,2009,22(5):704-708
- [8] 李娟,邵春福,杨励雅,等.基于Kalman滤波的行人跟踪方法研究[J].交通运输系统工程与信息,2009,9(6):148-153
- [9] Felzenszwalb P F, Girshick R B, McAllester D. Discriminatively trained deformable part models [C]//IEEE Conf. on Computer Vision and Pattern Recognition. 2009
- [10] Viola P, Jones M J, Snow D. Detecting Pedestrians Using Patterns of Motion and Appearance[J]. International Journal of

- [11] Pavan M, Pelillo M. Dominant sets and pairwise clustering [J]. PAMI, 2007, 29(1):167-172
- [12] Ge J, Luo Y, Tei G. Real-time pedestrian detection and tracking at nighttime for driver-assistance systems [J]. Intelligent Transportation Systems, IEEE Transactions on, 2009, 10(2):283-298
- [13] Geismann P, Schneider G. A two-staged approach to vision-based pedestrian recognition using Haar and HOG features [C]// Intelligent Vehicles Symposium, 2008 IEEE. IEEE, 2008: 554-559

(上接第 187 页)

行到第 8 行中遍历所有节点 N , 定义在模态窗口内的系统交互事件(S)和终止事件(T)分别为 S_M 和 S_T ; 第 9 行和第 10 行是删除 EFG 中所有的边; 第 11 行到第 13 行是遍历所有的节点 N , 去掉其中所有的可达事件(R)。第 14 行到第 17 行遍历所有的节点并令其自由交互生成边, 去掉边 (n_x, n_y) , 如果 n_x 不是 S_M 且 n_y 不是 S_M 或 T_M 。由于事件的驱动性, 模态窗口中的事件只能和本窗口内的事件交互, 因此模态窗口内的非终止事件与另一窗口或菜单内的事件交互是错误的, 需要移除这些事件序列。从图 1 到图 5 再到图 4 可以看到具体的转换过程。

驱动算法对应的 EIG 转换准则如下:

- (1)依次去掉 EFG 中的可达事件(R)和所有的边 (n_x, n_y) ;
- (2)对剩余的节点任取两个进行有序排列, 生成边 (n_x, n_y) (x, y 可以相等), 如图 5 所示;
- (3)去掉边 (n_x, n_y) , 如果 $n_x = S_M, n_y \neq S_M$ 或 $n_y \neq T_M$, 即去掉图 5 中有标号的 5 条边得到图 4;
- (4)对(1)中去掉的边 (R, n_y) 存储成映射 (R, n_y) , 用于在测试阶段执行; 共有 16 个映射:(编辑;复制)、(编辑;粘贴)、(编辑;中文简繁转换)、(编辑;定位)、(中文简繁转换;转换方向)、(中文简繁转换;确定)、(中文简繁转换;取消)、(定位;输入)、(定位;下一处)、(定位;关闭)、(定位;语言)、(定位;转换方向)、(定位;确定)、(定位;取消)、(定位;复制)、(定位;粘贴)。

3.3 MX 算法与驱动算法的比较

MX 算法和驱动算法都是由 EFG 转换成 EIG。但是, 驱动算法的优越性体现在两个方面:

(1)对于 EFG 转换成 EIG 过程中 GUI 的事件划分, 驱动算法比 MX 算法更详细。因为本文对 GUI 中的事件进行了更加全面详细的划分, 在驱动算法中使用了本文的 GUI 事件划分方式。

(2)对于 EFG 转换成 EIG 的过程, 驱动算法比 MX 算法更简单。从实例的转换过程图可看到, 即从图 1 到图 4, 在 MX 算法中去掉可达事件和与其相连的边后, 重新连接各个节点、生成新边的过程很繁杂, 容易出错和遗漏。而在驱动算法中就没有这一繁杂的过程, 这个算法中去掉了 EIG 中不具有的可达事件和所有的边, 然后对其中的所有事件进行两两排列配对(也可以与自身配对), 最后移除其中不正确的边。由实例可知, 对所有事件进行简单的排列配对(如图 5)比逐

个去掉可达事件及对与可达事件相连的事件根据跟踪关系重新配对(如图 2、图 3)要简单很多。

结束语 在基于模型的 GUI 测试技术中, EIG 是一种简单、高效的 GUI 模型。在 EFG 转换成 EIG 时, 本文对 GUI 中的事件进行了更加全面详细的划分, 并改进了原有的 MX 算法, 提出了驱动算法, 使得 EFG 生成 EIG 时更加简单。但是由于实验的局限性和为了有效地分析 GUI 模型, 本文只对一个菜单下的事件情况进行了研究, 有待于进一步提高。在未来的工作中我们会考虑不同菜单事件下的交互情况, 以及在生成 GUI 测试用例时可能不需要表述所有的 GUI 事件关系, 而是去掉中间的 EFG, 由 GUI 直接生成 EIG, 以便直接生成有效的测试用例。

参考文献

- [1] Memon A M. GUI Testing: Pitfalls and Process [J]. IEEE Computer, 2002, 8(35):90-91
- [2] Mathur A P. Foundations of Software Testing: Fundamental Algorithms and Techniques[M]. New Delhi: Pearson Education in South Asia, 2008:33-384
- [3] Memon A M. Advances in Computers[M]. Amsterdam, Netherlands: Elsevier Ltd, 2003:150-203
- [4] Memon A M, Xie Qing. Using Transient/ Persistent Errors to Develop Automated Test Oracles for Event-Driven Software[A]// Proceeding of 19th IEEE International Conference on Automated Software Engineering IEEE INFOCOM[C]. 2004:186-195
- [5] Memon A M. A Comprehensive Framework for Testing Graphical User Interfaces[D]. Pittsburgh: Department of Computer Science, University of Pittsburgh, 2001
- [6] Yuan Xun, Myra B C, Memon A M. GUI Interaction Testing: Incorporating Event Context [J]. IEEE Transactions on Software Engineering, 2011, 37(4):559-574
- [7] Memon A M, Xie Qing. Studying the Fault- Detection Effectiveness of GUI Test Cases for Rapidly Evolving Software[J]. IEEE Transactions on Software Engineering, 2005, 31(10):884-896
- [8] Xie Qing, Memon A M. Using a Pilot Study to Derive a GUI Model for Automated Testing[J]. ACM Transactions on Software Engineering And Methodology, 2008, 18(2):1-35
- [9] Brooks P A, Robinson B P, Memon A M. An Initial Characterization of Industrial Graphical User Interface Systems[A]// Proceeding of First IEEE International Conference on Software Testing Verification and Validation IEEE INFOCOM [C]. Denver, CO: IEEE, 2009:11-20