

基于 TLD 和 DTW 的动态手势跟踪识别^①

余超, 关胜晓

(中国科学技术大学 信息科学技术学院, 合肥 230027)

摘要: 针对动态手势跟踪稳定性的不足和识别效率的问题, 提出一种基于 TLD 和 DTW 的动态手势跟踪识别框架. 首先利用基于 Haar 特征的静态手势分类器获得手势区域, 然后使用 TLD 跟踪算法对获得的手势区域进行跟踪以获取手势轨迹, 最后提取轨迹特征, 使用改进的 DTW 算法进行识别. 实验表明, 该框架能够长时间稳定地跟踪手势区域, 并能够在保证识别率的基础上显著提高识别效率.

关键词: 动态时间规整; 手势跟踪; 手势识别; TLD; Haar

Dynamic Hand Gesture Tracking and Recognition Based on TLD and DTW

YU Chao, GUAN Sheng-Xiao

(School of Information Science and Technology, University of Science and Technology of China, Hefei 230027, China)

Abstract: For lack of stability in dynamic hand gesture tracking and low recognition efficiency, a hand gesture tracking and recognition framework is proposed based on TLD and DTW algorithm. First, the hand gesture area is got with a Haar features based classifier. Then the gesture trajectory is obtained by using the TLD tracking algorithm initialized by the hand area. Finally, features are extracted from the trajectory and the improved DTW algorithm is used to recognize the dynamic gesture. Experiments show that the framework can track the hand gesture stably for a long time and be able to improve the recognition speed greatly under the premise of ensuring high recognition accuracy.

Key words: dynamic time warping (DTW); hand gesture tracking; hand gesture recognition; tracking-learning-detecting (TLD); Haar

1 引言

手势作为人机交互(Human Computer Interaction, HCI)接口的一种, 具有以人为中心, 简单、高效、人性化的特点, 是人们研究人机交互的重要手段之一.

经典的手势跟踪算法^[1-3]中, 均值漂移(Meanshift)算法不能够很好地适应目标尺寸变化和目标的形变; 连续自适应均值漂移(Camshift)算法能够很好地利用颜色信息进行跟踪, 但是容易受到相同或相近颜色的干扰; 卡尔曼滤波(Kalman Filter)算法的假设条件可能不成立; 粒子滤波(Particle Filter)算法在使用过程中容易出现粒子数目匮乏和粒子多样性的问题, 导致跟踪失败. 文献[4]在 Meanshift 算法的基础上引入 Kalman 滤波器来预测目标的位置, 虽然在一定程度上解决了 Meanshift 算法要求两帧间目标模板必须重叠

的缺陷, 提升了跟踪的成功率, 但跟踪的准确性有待提高. 文献[5]用似然函数来降低粒子滤波算法中重要性权值的方差, 提高了抽样效率, 跟踪准确性也有所提高, 但算法复杂度较高. 文献[6]提出了 TLD(Tracking-Learning-Detection)算法, 利用在线学习机制解决了长时间跟踪的稳定性问题, 但需要通过手动框选的方式来指定目标的初始化位置, 框选位置的不同对跟踪的结果有较大的影响, 会导致跟踪不稳定.

手势识别算法^[7]中常用的方法有动态时间规整(Dynamic Time Warping, DTW)和隐马尔可夫模型(Hidden Markov Model, HMM). DTW 只需要少量的模板就能够对样本进行匹配, 但算法的时间和空间复杂度较高. HMM 训练目标模型需要大量的样本, 且初始参数的选择对最终模型的识别率及运算的复杂度有

^① 收稿时间:2015-02-03;收到修改稿时间:2015-04-07

较大影响. 文献[8]通过实验对比 DTW 和 HMM 算法在手势识别中的应用, 说明了 DTW 算法有更好的表现.

为了解决跟踪的稳定性问题, 本文在 TLD(Tracking-Learning-Detection)算法的基础上, 改进其初始化策略, 通过确定合适的初始化位置, 实现了手势的长时间稳定跟踪. 同时改进 DTW 算法, 在保证识别率的基础上降低 DTW 算法的复杂度, 进一步提升识别效率.

本文提出的基于 TLD 和 DTW 的动态手势跟踪识别框架如图 1 所示. 首先利用基于 Haar 特征的静态手势分类器获得手势区域, 然后使用 TLD 跟踪算法对获得的手势区域进行跟踪以获取手势轨迹, 最后提取轨迹特征, 使用改进的 DTW 算法进行识别.

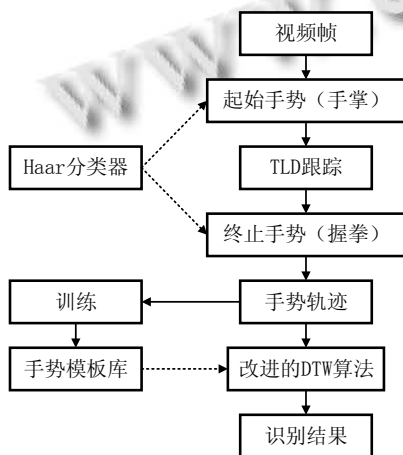


图 1 基于 TLD 和 DTW 的动态手势跟踪识别框架图

本文后续节从 Haar 分类器的构建、TLD 跟踪算法及其改进、DTW 算法及其改进、手势模板库的构建、手势识别及实验结果分析等方面阐述改进的框架的原理及效果.

2 Haar分类器

构建 Haar 分类器的主要目的是为了进行手势轨迹起始点(手掌手势)和终止点(握拳手势)的检测. Haar 分类器的构建主要步骤包括提取 Haar 特征和训练分类器两部分.

2.1 提取 Haar 特征

Haar 特征主要分为四类: 边缘特征、线性特征、中心特征和对角线特征, 典型的四个 Haar 特征模板如图 2 所示. 将特征模板在样本图像上滑动, 用模板位

置白色区域的像素和减去黑色区域的像素和便可以得到一个特征值, 这些特征值就是弱分类器训练的基础.

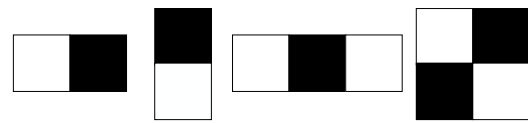


图 2 Haar 特征

2.2 训练分类器

本文采用改进后的 Adaboost^[9]算法来训练分类器. 算法首先从样本提取的 Harr 特征训练不同的分类器作为弱分类器, 然后把把这些弱分类器集合起来, 构成一个最终的强分类器.

Adaboost 算法的具体流程如下:

假设 X 表示样本空间, Y 表示样本类别标识集合, 对于两分类问题 $Y = \{0, 1\}$, 令 $S = \{(x_i, y_i) \mid i = 1, 2, 3, \dots, m\}$ 为加入标签后的训练样本集合, 其中 $x_i \in X, y_i \in Y$, 假设一共迭代了 T 次.

① 初始化 m 个样本的权值 $D_1(i) = 1/m, i = 1, 2, 3, \dots, m$, 记 $D_t(i)$ 表示在第 t 次迭代中赋给样本 (x_i, y_i) 的权值

② $t = 1, 2, 3, \dots, T$

a. 对样本 x 的每一个特征 f 训练一个弱分类器

$$h_t(x, f, p, \theta) = \begin{cases} 1 & pf(x) < p\theta \\ 0 & otherwise \end{cases}$$

其中, θ 为特征 f 对应的分类器的最优阈值, p 用来调整不等号的方向.

计算所有特征的弱分类器加权(q_i)后的分类错误率 ϵ_f .

$$\epsilon_f = \sum_i q_i |h_t(x, f, p, \theta) - y_i|$$

b. 选出最佳的弱分类器 $h_t(x)$ (拥有最小的错误率 ϵ_t)

$$\epsilon_t = \min_{f, p, \theta} \sum_i q_i |h_t(x, f, p, \theta) - y_i|$$

$$h_t(x) = h_t(x, f_t, p_t, \theta_t)$$

c. 根据这个最佳的弱分类器来修正权重.

$$D_{t+1}(i) = D_t(i) \beta_i^{1-\epsilon_i}$$

$$\beta_i = \epsilon_i / (1 - \epsilon_i)$$

其中, 如果 x_i 被正确分类, 则 $\epsilon_i = 0$, 否则 $\epsilon_i = 1$.

③ 最后的强分类器 $C(x)$ 表示如下

$$C(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & otherwise \end{cases}$$

$$\alpha_i = \log(1 / \beta_i)$$

本文借助上述方法分别训练了手掌手势(起始手势)和握拳手势(终止手势)分类器,实现了手势轨迹起始位置和终止位置的准确划分。

3 TLD跟踪算法及改进

3.1 标准 TLD 算法

TLD 算法是 Zdenek Kalal 提出的一种新的单目标长时间跟踪算法。该算法与传统跟踪算法的显著区别在于它将传统的跟踪算法和传统的检测算法相结合,解决了被跟踪目标在跟踪过程中发生形变、部分遮挡等问题。同时,通过一种改进的在线学习机制不断更新跟踪模块的“显著特征点”和检测模块的目标模型及相关参数,从而使得跟踪更加稳定、鲁棒和可靠。

TLD 算法框架如图 3 所示,主要由检测模块、跟踪模块和学习模块构成。检测模块在每一帧图像中独立进行目标检测。跟踪模块根据前一帧中的跟踪结果预测目标在当前帧中的位置。学习模块通过 PN-Learning 机制评估检测结果与跟踪结果,修正目标模型,使跟踪过程更加准确、稳定。

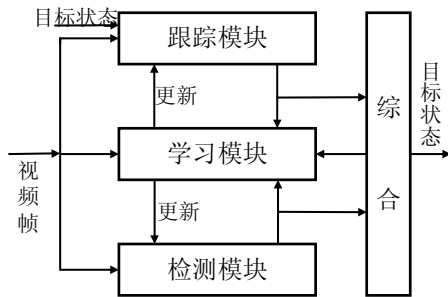


图 3 TLD 算法框架

3.2 TLD 算法的改进

标准的 TLD 算法采用手动框选的方式来确定跟踪目标,一方面,框选位置的不同会影响跟踪的稳定性;另一方面,不便于整个跟踪识别流程的自动化运行。本文引入 Haar 分类器来检测起始手势(手掌手势),并将检测到的区域用来初始化 TLD 跟踪器,触发跟踪过程,然后再检测终止手势(握拳手势)结束跟踪过程。

由于 Haar 分类器本身的效率不是很高,对手掌手势和握拳手势的检测进行了如下优化:

① 在检测起始手势前,利用 HSV 色彩空间的肤色分割算法[10]剔除视频帧中的非肤色区域,让检测

器在当前帧中的肤色区域(可能包含部分类肤色区域)中进行起始手势的检测,从而提高检测的效率;

② 设定进行终止手势检测的条件。当人手在某位置停留超过 10 帧时,触发终止手势的检测。检测时将 TLD 跟踪结果标定的当前手势位置矩形框长和宽分别扩大 10%作为搜索区域进行检测,一方面大大减少搜索范围,另一方面在一定程度上降低 TLD 跟踪矩形框出现的偏差对终止手势检测结果的影响。

4 DTW算法及改进

DTW 算法是 60 年代由日本学者提出的一种衡量两个长度不同的时间序列的相似度的方法。DTW 的应用范围比较广,比如说用在孤立词语音识别、手势识别、数据挖掘和信息检索等,图 4 展示了使用 DTW 算法对两个时间序列进行最佳匹配的示意图。

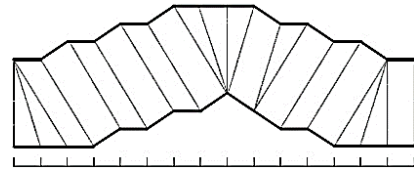


图 4 使用 DTW 算法匹配两个时间序列的示意图

4.1 标准 DTW 算法

DTW 基本原理如下:

假设有两个时间序列 $Q = \{q_1, q_2, \dots, q_i, \dots, q_n\}$ 和 $C = \{c_1, c_2, \dots, c_j, \dots, c_m\}$ 。

首先构造一个 $n * m$ 的矩阵,矩阵元素 (i, j) 表示分量 q_i 和 c_j 之间的距离 $dist(q_i, c_j)$,这里一般使用欧氏距离,记为 $dist(i, j)$ 。

定义累积距离 $D(i, j)$ 为当前格点距离 $dist(i, j)$ 与可以到达该点的最小的邻近元素的累积距离之和:

$$D(i, j) = dist(i, j) + \min \{D(i - 1, j), D(i, j - 1), D(i - 1, j - 1)\}$$

其中 $D(1, 1) = dist(1, 1)$.则累积距离 $D^{(n, m)}$ 就是序列 Q 和 C 的相似性度量。

4.2 改进的 DTW 算法

从标准 DTW 的计算公式可以看出,对于弯曲矩阵中的每个方格都计算,DTW 计算的时间复杂度为 $O(m*n)$ 。由于额外构造了弯曲矩阵,DTW 计算的空间复杂度为 $O(m*n)$ 。为了改进 DTW 算法,提高算法效

率, 学者们提出了很多改进算法, 如: FastDTW^[11]、SparseDTW、LB_Keogh^[12]、LB_Improved 和 LB_HUST 算法等。

时间序列相似度比较过程中, 距离度量是衡量相似的重要部分, 但是由于 DTW 具有较高的计算复杂度, 实际中往往不是直接计算原始 DTW 距离, 而是首先通过计算 DTW 的距离下界函数来快速剔除匹配度较低的模板, 只对剩下的模板计算序列间的 DTW 距离。

下界函数的选择应满足下面两个标准:

- ① 能快速计算, 最好能够在线性时间内完成。
- ② 下界距离函数应尽可能贴近实际的距离。

上面两个条件保证了使用下界函数筛选的有效性。

本文采用了下界函数 LB_HUST^[13], 其表示如下:

$$D_{lb_hust}(Q, C) = \sqrt{\sum_{i=1}^n (diff_i)^2}$$

$$diff_i = \begin{cases} (c_i^L - q_i^U) & c_i^L > q_i^U \\ (q_i^L - c_i^U) & q_i^L > c_i^U \\ 0 & otherwise \end{cases}$$

文献[13]在理论上证明了 LB_HUST 是 DTW 距离的下界函数, 且 LB_HUST 下界于 LB_Keogh, 与 DTW 的紧密性与 LB_Keogh 相当, 这里不再赘述。同时 LB_HUST 还满足对称性, 即

$$D_{lb_hust}(Q, C) = D_{lb_hust}(C, Q)$$

这是其他几种下界函数所不具有的。

5 手势模板库的构建

5.1 手势轨迹特征的提取

TLD 跟踪的结果可转换为一个手势中心位置坐标的序列 P, 考虑到实验者做手势时手的抖动以及跟踪算法本身的误差, 需要先对序列 P 进行平滑处理, 从而在一定程度上消除误差。

假设序列 $P = \{p_1, p_2, \dots, p_l\}$, 滑动窗口大小为 w , 步长为 $s, l < s \leq w$, 则平滑后的序列为

$$P^* = \{p_1^*, p_2^*, \dots, p_n^*\}$$

满足:

$$p_i^* = \sum_{j=1+(i-1)*s}^{(i-1)*s+w} p_j / w$$

其中, n 为满足下面条件的最大整数:

$$(n-1) * s + w \leq l, \text{ 即 } n \leq (l-w) / s + 1$$

所以 $n = \lfloor (l-w) / s \rfloor + 1$, $\lfloor x \rfloor$ 表示对 x 向下取整。

对于平滑后的点序列 P^* , 如图 5 所示, 计算相邻两点连线与 x 轴正方向的夹角 θ , 并离散化到图示的八个方向。

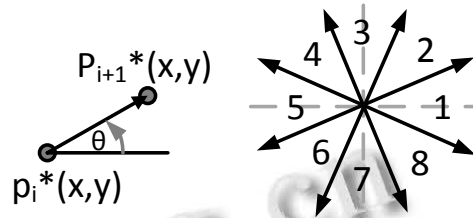


图 5 提取手势运动方向特征

经过上述处理后就获得了手势方向特征序列 $F = \{f_1, f_2, \dots, f_m\}, m = n - 1$ 。

5.2 获取最佳手势模板

对于动态手势 $a, a \in A$ (A 为所有预定义的动态手势类型的集合) 的 n 个样本, 分别提取方向特征序列得到方向特征序列集合 $F^* = \{F_1, F_2, \dots, F_n\}$

定义评价函数

$$g(F_i) = \sum_{j=1}^n D(F_i, F_j)$$

其中, $D(F_i, F_j)$ 表示序列 F_i 和 F_j 间的动态时间规整距离。

获取手势 a 的最佳模板的过程就是从手势 a 的 n 个样本中找到一个样本 i , 使得对 $\forall i \in \{1, 2, 3, \dots, n\}$, 有

$$g(F_i) \leq g(F_j)$$

这样就获得了手势 a 的最佳模板为样本 i , 同理可以获得其他动态手势的最佳模板。

6 动态手势的识别

已知待识别手势序列 C , 手势模板库 $T = \{T_1, T_2, \dots, T_n\}$, 手势序列 C 的识别过程就是从模板库中找到与 C 最佳匹配的模板 $T_i, i \in \{1, 2, 3, \dots, n\}$, 识别过程包括快速识别和精确识别两个阶段。

① 快速识别

DTW 算法本身计算的复杂度很高, 随着手势模板库中手势的增加, 识别的效率会急剧下降。快速识别的目的是通过复杂度低的下界函数算法快速剔除匹配可能性低的模板, 从而减少精确匹配时的计算量。

本文首先使用 LB_HUST 下界函数快速计算 C 与模板 $T_i, i \in \{1, 2, 3, \dots, n\}$ 的下界距离 $D_{lb_hust}(T_i, C)$, 设定阈值 θ , 如果

$$D_{lb_hurst}(T_i, C) > \theta$$

则从匹配列表中移除模板 T_i 。如此可以将大量匹配度低的模板从匹配列表中移除, 获得筛选后的匹配列表, 供后续精确匹配使用。

② 精确识别

通过快速识别获得筛选后的匹配列表, 大大减少了待匹配的模板的数量。从剩下的待匹配模板中计算与待匹配序列动态规整距离最小的模板作为最终的识别结果。相对于标准的 DTW 算法大大降低了运算量, 在保证识别率的基础上能有效地提高识别效率。

7 实验结果与分析

7.1 Haar 分类器的性能

实验分别训练了手掌手势和握拳手势分类器。训练样本一部分来自 INRIA 数据库(INRIA Person Dataset)^[14], 一部分是自己录制。其中训练手掌手势分类器时正样本 1233 个, 负样本 5346 个。训练握拳手势分类器时正样本 1136 个, 负样本 5324 个。

表 1 显示了分类器的性能参数, 其中测试视频(分辨率 320x240)总长 459 帧, 由手掌手势(110 帧)、握拳手势(105 帧)、其他手势(96 帧)和非手势(148 帧)几部分构成。

表 1 Haar 分类器性能

分类器类型	检出率(%)	虚警率(%)	时间性能(fps)
手掌分类器	90.1	7.3	15.2
握拳分类器	92.4	8.1	16.4

从表 1 中可以看出, 与传统基于肤色分割和模板匹配(识别率 50%~80%左右, 处理速度 20~25fps 左右)的方法相比, Haar 分类器牺牲了部分时间性能, 但获得了较高的识别率, 能保证传递给 TLD 跟踪算法的初始跟踪位置的正确性。但由于同时存在虚警, 所以检测手势时常采用多次检测以超过半数的检测结果来确认最终的手势类型。

7.2 TLD 跟踪算法性能

实验数据参照 SKIG 手势数据库^[15]分别录制了保持手掌手势和握拳手势运动的视频, 并手动标记了手掌和拳头的真实位置用来评估 TLD 算法性能并和 Struck 算法^[16,17]进行了对比。

文中采用重叠度来评价跟踪的有效性, 重叠度定义如下:

$$overlap = \frac{W_i \cap W_{gi}}{W_i + W_{gi} - W_i \cap W_{gi}}$$

其中, W_i 表示跟踪算法获得的第 i 帧图像中目标位置矩形框, W_{gi} 表示第 i 帧图像中目标的真实位置矩形框, $W_i \cap W_{gi}$ 表示跟踪结果和真实位置相交区域。

图 6 所示的是 TLD 跟踪结果(红色虚线)、Struck 跟踪结果(蓝色虚线)与真实位置(黑色实线)的对比。

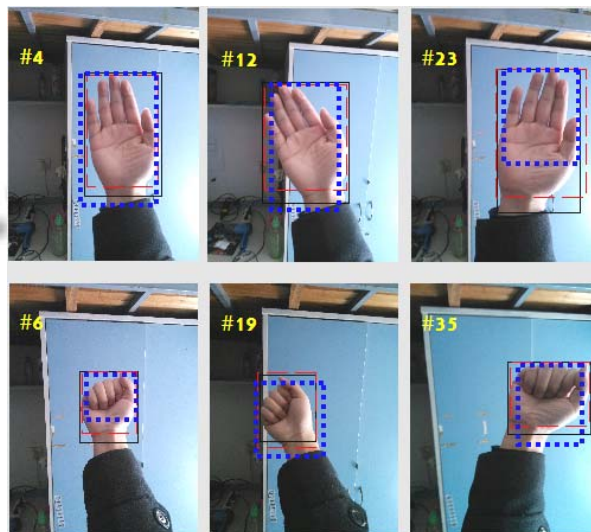


图 6 TLD、Struck 算法跟踪结果与真实位置对比

实验结果中, 认为才为有效跟踪, 两种算法对比结果如表 2 所示。

表 2 TLD 跟踪算法性能

跟踪算法	跟踪率(%)	有效跟踪(%)	时间性能(fps)
TLD	93.1	90.7	16.3
Struck	89.5	86.5	12.7

从表 2 中可以发现 TLD 算法的跟踪率较高, 处理的实时性也较好, 能更稳定的跟踪目标(有效跟踪率较高)。同时注意到 TLD 跟踪中无效的跟踪也占了一定的比例, 所以在提取轨迹特征前对轨迹使用滑动窗口进行平滑是很有必要的, 可以在一定程度上消除跟踪误差对识别带来的影响。

7.3 改进的 DTW 算法的性能

实验中设计了如图 7 所示的 10 种手势。由 4 个实验者在 3 种光照条件下每种手势做 4 次, 利用 TLD 跟踪算法共得到 480 条手势轨迹, 提取特征后, 得到 480 个手势运动方向特征序列, 使用其中的 240 条进行训练得到 10 种手势的模板, 剩余的 240 个作为测试样本。

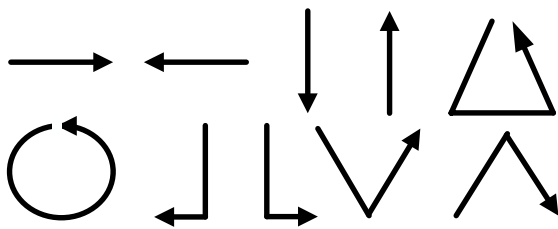


图 7 预定义手势类型

7.3.1 快速识别

快速识别阶段采用 LB_HUST 下界函数来加速识别算法计算过程, 需要确定合适的阈值 θ 来保证正确的模板在快速识别阶段保留下来, 以确保精确识别阶段能正确识别手势. 图 8 展示了快速识别阶段留下的模板数目(阈值)对最终识别结果的影响.

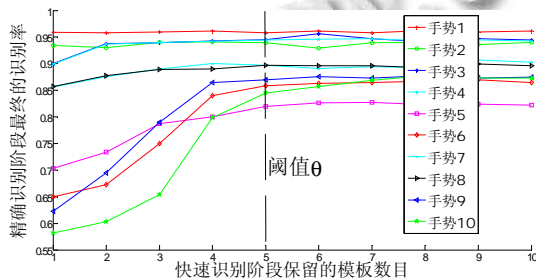


图 8 快速识别阶段阈值 θ 的选择

图 8 表明, 针对手势 1、2、3、4、7、8, 只需要在快速识别阶段保留最匹配的两个模板就能保证精确识别阶段的识别率. 但是对于手势 5、6、9、10 需要至少保留五个最匹配的模板才能保证快速识别阶段的识别率. 为了保证精确识别阶段的识别率, 在快速识别阶段本文采取了统一保留 LB_HUST 距离最小的五个模板作为精确识别阶段的匹配模板. 这样虽然没有能够充分的发挥出快速识别阶段对整体识别过程效率的提升, 但是保证了最终的整体识别率.

7.3.2 精确识别

实验对 240 个测试样本使用标准 DTW 算法及改进的 DTW 识别算法进行了 10 次重复实验得到 DTW 算法改进前后的平均性能指标如表 3 所示.

表 3 改进的 DTW 算法性能

手势类型	平均识别率(%)		时间性能(s)	
	标准	本文	标准	本文
	DTW	方法	DTW	方法
手势 1	95.8	96.1	0.15	0.14

手势 2	94.3	94.0	0.19	0.17
手势 3	94.7	94.4	0.36	0.08
手势 4	94.8	94.3	0.18	0.09
手势 5	85.7	82.2	0.29	0.13
手势 6	87.9	86.5	0.65	0.40
手势 7	90.0	90.3	0.24	0.30
手势 8	91.5	89.6	0.22	0.17
手势 9	89.9	87.5	0.34	0.24
手势 10	88.3	87.3	0.56	0.36

从表 3 可以看出, 虽然为了保证最终的识别率, 在快速识别阶段筛选保留了 LB_HUST 距离最小的五个模板来进行精确识别, 在一定程度上限制了效率的提升. 但手势 3、4、5、6、9、10 识别效率的提升还是比较明显的, 并且识别率降低最大不超过 3%, 实现了文中的目标. 对于手势 1、2、7、8 在快速识别阶段原本保留两个模板就能保证最终的识别率, 由于文中保留了五个, 导致效率提升不明显, 后续需要继续研究如何动态决定快速识别阶段保留的模板的数目来继续提升效率.

8 结语

本文主要针对手势跟踪的稳定性问题和识别的效率问题, 提出一种基于 TLD 和 DTW 的手势跟踪识别框架. 实验结果表明, 该框架借助 TLD 算法能有效地实现对手势稳定地跟踪, 且改进的 DTW 识别算法在保证识别率的基础上, 有效地提高了识别效率, 部分手势识别效率提升超过了 50%. 文中在动态手势的快速识别阶段, 为了保证部分手势的识别率, 增加了保留模板的数目, 这在一定程度上限制了其他手势识别效率的提升, 如何动态自适应地确定快速识别阶段应保留的模板数目, 进一步提升识别的效率是本文下一步要研究的方向.

参考文献

- 1 武霞, 张崎, 许艳旭. 手势识别研究发展现状综述. 电子科技, 2013, 26(6): 171-174.
- 2 彭艳芳. 视频运动目标检测与跟踪算法研究[硕士学位论文]. 武汉: 武汉理工大学, 2010.
- 3 高文, 朱明, 贺柏根, 吴笑天. 目标跟踪技术综述. 中国光学, 2014, 7(3): 365-375.
- 4 刘继艳, 潘建寿, 吴亚鹏, 王宾, 付勇. 结合 Kalman 滤波器的 Mean-Shift 跟踪算法. 计算机工程与应用, 2009, 45(12):

- 184-186,197.
- 5 何广.基于粒子滤波的目标跟踪算法研究[硕士学位论文]. 哈尔滨:哈尔滨工业大学,2006.
- 6 Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2012, 34(7): 1409-1422.
- 7 Fang CS. From Dynamic Time Warping (DTW) to Hidden Markov Model (HMM). Final Project Report for ECE742 Stochastic Decision. University of Cincinnati, 2009.
- 8 Carmona J, Climent J. A performance evaluation of HMM and DTW for gesture recognition. *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, 2012: 236-243.
- 9 季怡,杨帆,龚声蓉,刘纯平.基于 Adaboost 实现的实时手势识别.实验室研究与探索,2014,33(8):123-126,139.
- 10 田巍,庄镇泉.基于 HSV 色彩空间的自适应肤色检测.计算机工程与应用,2004,14:81-85.
- 11 Salvador S, Chan P. FastDTW: toward accurate dynamic time warping in linear time and space. *KDD Workshop on Mining Temporal and Sequential Data*. 2004.
- 12 Keogh E. Exact indexing of dynamic time warping. *Knowledge and Information Systems*, 2005, 7(3): 358-386.
- 13 李俊奎.时间序列相似性问题研究[博士学位论文].武汉:华中科技大学,2008.
- 14 INRIA Person Dataset. <http://pascal.inrialpes.fr/data/human/>.
- 15 SKIG 手势数据库. <http://lshao.staff.shef.ac.uk/data/SheffieldKinectGesture.htm>.
- 16 Hare S, Saffari A, Torr PHS. Struck: structured output tracking with kernels. *2011 IEEE International Conference on Computer Vision (ICCV)*. IEEE. 2011. 263-270.
- 17 Wu Y, Lim J, Yang MH. Online object tracking: A benchmark. *2013 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2013. 2411-2418.