Computer Science and Application 计算机科学与应用, 2016, 6(4), 248-264
Published Online April 2016 in Hans. http://www.hanspub.org/journal/csa
http://dx.doi.org/10.12677/csa.2016.64031

Hans汉斯

The Automatic Object Detection System Based on TLD Framework

Xueyan Li, Chunnan Wang, Min Xie, Changdong Wang

School of Data and Computer Science, Sun Yat-sen University, Guangzhou Guangdong
Email: leexueyan@foxmail.com, yefufeng@foxmail.com, 1784108956@qq.com, changdongwang@hotmail.com

Received: Apr. 8th, 2016; accepted: Apr. 26th, 2016; published: Apr. 29th, 2016

Abstract

With the enhancement of the data processing ability of computer, the technology on sensor, audio and automation control has been developed continuously, and the information in video frames and image has got a lot of attention, which is one of the main sources that human obtain information from the world. Computer vision, as one of the present research upsurges, has many technical challenges such as detection, motion, scene reconstruction and image restoration. Object detection is one of the most important challenges. Although there are plenty of object detection systems with high accuracy rate of detection in the market, they lack realization on auxiliary functions so that they provide poor experience on man-machine interaction. Therefore, many developers focus on the topic that how to design a better man-machine interaction of detection system for human so that the detection system can be accepted widely. In this paper, we propose a system framework which contains the technology on object detection and voice processing. Firstly, we make improvement on the algorithm of Tracking-Learning-Detection (TLD). We use the image sets of the object which we want to detect to get a suitable classifier by training algorithm. Then, we can use the classifier to determine whether the new object is the target object and get the aim of detecting the specified object. Then, the system contains the module of speech recognition for a better man-machine interaction so that the user can add the image data to the data set and update the classifier by voice. In order to guarantee the accuracy of speech recognition, we use the Dynamic Time Warping (DTW) to match the phonetic characteristics.

Keywords

Object Detection, Voice Processing, Tracking-Learning-Detection, Dynamic Programming Algorithm

文章引用: 李学彦, 王春南, 谢敏, 王昌栋. 基于 TLD 的物体自动识别系统[J]. 计算机科学与应用, 2016, 6(4): 248-264.
http://dx.doi.org/10.12677/csa.2016.64031

基于TLD的物体自动识别系统

李学彦，王春南，谢　敏，王昌栋

中山大学数据科学与计算机学院，广东 广州
Email: leexueyan@foxmail.com, yefufeng@foxmail.com, 1784108956@qq.com, changdongwang@hotmail.com

收稿日期：2016年4月8日；录用日期：2016年4月26日；发布日期：2016年4月29日

摘　要

大数据时代下随着计算机数据处理能力的提高，传感技术、音频技术、自动化控制技术得到不断地发展，视频帧和图像信息作为人类通过客观世界获得信息的主要来源之一更是得到了诸多的重视。如今计算机视觉作为当下研究的热潮之一，拥有诸如识别、运动、场景重建、图像恢复等众多技术挑战。其中又以物体识别最为重要。与此同时，众多的物体识别系统却仅仅侧重于物体识别的精度而缺乏其他辅助功能的实现，如何拥有更好的人机交互以及更广阔的市场前景的物体自动识别系统是当下众多开发者所探讨的。在本文中我们将物体识别与语音处理相结合，首先在物体识别算法Tracking-Learning-Detection (TLD)的基础上进行改进，以给定的一类物体的图片数据集为基础，训练出适合于识别该类物体的分类器，从而判断新的物体是否为目标物体，实现对指定一类物体的识别；同时该系统将以语音识别作为人机交互的基础，使用户可以利用语音将图片数据添加到训练集中并更新分类器，同时采用动态规划的方式(DTW)对语音特征进行匹配从而保证了语音识别的准确度。

关键词

物体识别，语音处理，TLD，DTW

1. 介绍

如今，计算机视觉的研究不论在国内或国际的期刊及会议依旧处于热潮之中，在这一过程中有许多经典的物体识别的算法得到了更好的优化和改进，但其依旧有自身的缺点和不足。如帧间差分法对差分的帧的选择时机要求很高，若物体运动过快导致帧之间的时间间隔过大就会很难分割出目标物体[1]；而背景差分法[2] [3]则对光照条件变化敏感，从而影响检测结果的准确性；光流法[4]的复杂程度和耗时的缺陷就更加难以满足实时检测的要求。传统的识别模型和算法已经无法适应当下具有高运算和数据处理能力的时代。随着视觉测试标准的逐步确立，出现越来越多的开放的测试基准平台，如 VTB、VOT [5] [6]，通过光照变化、尺寸变化、遮挡、形变、运动模糊、平面转动、相似目标及低分辨率等评估方法来评测各种算法对不同场景的处理能力[7]，其中 TLD 算法在众多综合测评中更优。在该算法的基础上进行改进后就需要使系统更加智能化：根据用户反馈的结果对物体的分类器更新，并辅以语音处理的功能让用户以语音的方式将结果反馈给系统。

1.1. 相关工作

由于早期的计算机运算和处理能力有很大的局限性，在图像处理方面，仍停留在静态图像的处理阶段，其工作主要集中在二维图像分析和识别上，如光学字符识别，工件表面、显微图片和航空图片的分

析和解释等。60 年代，Robert 通过计算机从数字图像中提取出诸如立方体、楔形体、棱柱体等多面体的三维结构，并对物体形状及物体空间关系进行了描述[8]，从这开始，研究的范围从边缘、角点等特征提取，到线条、平面、曲面等几何要素分析，一直到图像明暗、纹理、运动以及成像几何等，已经出现了一些视觉应用系统并建立了各种数据结构和推理规则[9]。在 1998 年，Benjamin Coifmana 和 David Beymerb 等人将视觉跟踪分为四类，分别是基于区域的跟踪、基于特征的跟踪、基于变形模板的跟踪和基于模型的跟踪[10]，目前包括 TLD 在内的大多主流的视觉跟踪算法都被囊括其中。

本文的 TLD 算法将长期跟踪任务分解为三个子任务：跟踪、检测和学习。跟踪器可以一帧帧的不断跟踪；检测器则对目标物体进行局部化处理，并根据需要修正跟踪器的错误；学习器则估计出检测器的错误并及时更新检测器避免重复的错误。每个子任务有单独部件处理，所有部件会在同时一起运行[11]。同时结合语音识别和合成的二次平台 Microsoft speech SDK，让机器接收、识别和理解语音信号，并将其转换成相应的数字信号，使用户可以用声音来代替鼠标和键盘完成部分操作，实现人机交互的效果。本文将基于 TLD 算法并以语音为辅实现物体自动识别系统。

1.2. 我们的工作

研究内容主要包括两大部分：物体识别和语音处理。系统繁荣总体框架如图 1 所示。物体识别又可以分为目标跟踪、目标检测和机器学习：其中目标跟踪和目标检测可以合为图像特征提取的过程，即先对目标的运动进行估计，然后定位输入图像中的目标物体，对目标物体的图像特征进行提取，即通过运算来检查图像的每个像素并确定图像所处的特征；机器学习的目的是把数据转换成信息[12]。

语音处理则包括语音识别和语音合成两部分：语音识别包括选择语音识别单元的语音信号预处理与特征提取以及通过获取的语音特征所使用的训练算法进行训练后产生的声学模型与模式匹配两个部分；语音合成则是构建包括由识别语音命令构成的语法网络或由统计方法构成的语言模型，并最终利用语法和语义分析对语言进行处理。语音识别的流程如图 2 所示。

2. 物体自动识别系统的实现

系统的实现需要 TLD 算法和 Speech SDK 相结合，前者是完成核心的物体识别的功能，后者将作为人机交互的功能实现的平台。

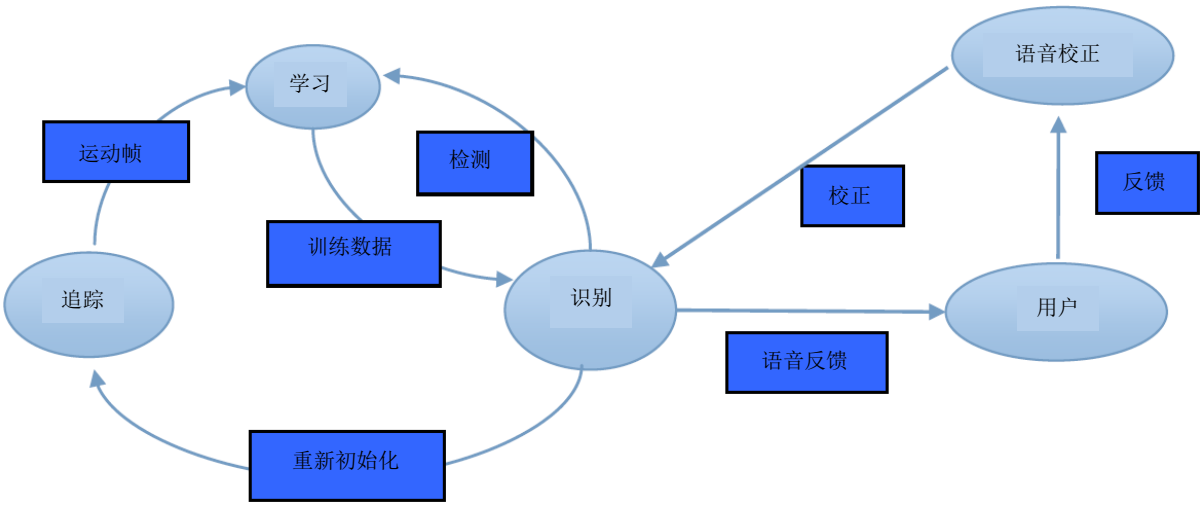


Figure 1. Framework of the automatic object detection system
图 1. 物体自动识别系统框架

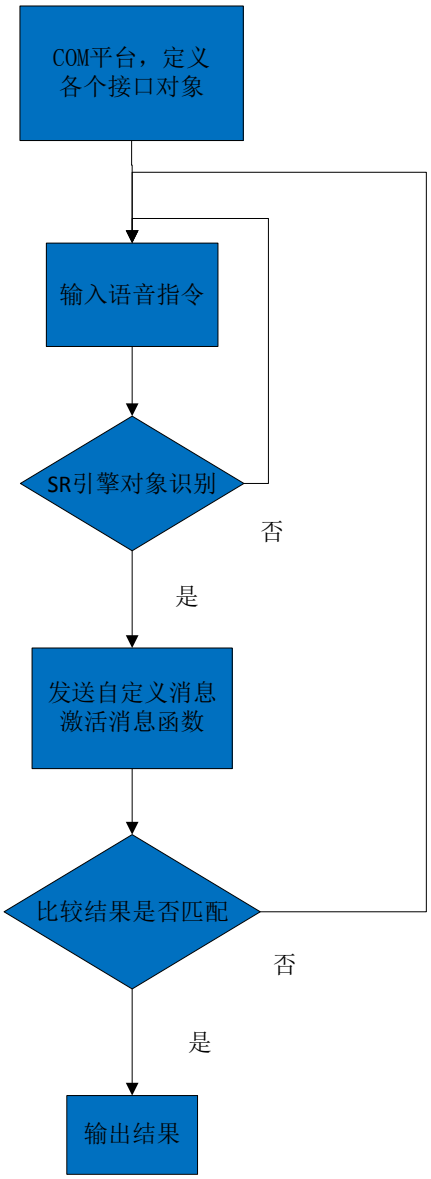


Figure 2. The flowchart of speech recognition
图 2. 语音识别流程

2.1. 物体识别

根据 TLD 算法框架，假设图像与图像之间的移动距离有限且目标物体仍在计算机的视觉区域内，追踪器用来估算两幅图像之间目标物体的运动；识别器通过扫描目标图像，并观察与学习到的数据来定位目标物体的位置；由于识别器在进行物体识别时可能产生错误，学习部分通过观察追踪器和识别器的性能来估算识别器产生的误差，生成新的训练样例来避免以后产生相同的错误。在学习部分的作用下，识别器可以生成更多的对象特征以将目标物体与背景区分开。

2.1.1. 追踪器

在追踪器中，使用基于带错误检测的 Midian-Flow 追踪算法。Midian-Flow 追踪算法通过边界框来表示目标物体并且估计它在连续图像之间的运动。追踪器通过估计对象边界框中点的位移来估算它们的可

靠性，然后选举出其中最可靠的50%的点。该算法采用的是Lucas-Kanade追踪器，只需要知道给定的若干个追踪点，追踪器会根据这些像素的运动情况确定这些追踪点在下一帧的位置。

追踪点的确定则需要先在上一帧的物体周围框产生均匀点，同时采用Lucas-Kanade追踪器追踪这些点直到下一帧，然后再反向追踪到上一帧，计算FB误差并筛选出FB误差最小的一半点作为最佳追踪点。最后根据点的距离和坐标变换计算下一帧的框图的大小和位置。效果如图3所示。

在检测错误时 Midian-Flow 追踪算法假设所有的物体都是可见的，这样的话当物体移动到视野外面时，就会产生检测错误，从而退化追踪器的性能。为了标识出这种情况，令 d_i 为 Midian-Flow 追踪器中单个点的位移，d_m 为位移的中值，当 $|d_i - d_m|$ 值大于 10 个像素时，就认为追踪器产生了一个检测错误。通过这种策略，可以在物体高速移动的情况下检测到错误。当错误被检测到时，追踪器就不会产生任何的边界框。

2.1.2. 识别器

识别器主要是由方差滤波器、集成分类器和最近邻分类器组成的级联分类器，用来识别目标物体在图像中的位置，确定追踪点的流程如图4所示。

1) 候选区域为图像中目标可能出现的任意区域，候选区域只有通过了方差滤波器、集成分类器和最近邻分类器的判断才会被判定为目标区域，否则就会被认为该区域不存在目标物体。

2) 方差滤波器通过计算候选区域的方差与原有目标区域的方差，并将其作比较，把方差小于原目标区域方差一半的候选区域排除。一个区域 p 的灰度值方差可以用 $IE\left(p^2\right)-IE^2\left(p\right)$ 来表示，其中期望值 $IE\left(p\right)$ 可以由恒定时间内使用集成图像来测得。通过这种方式，可以大大减少对图像区域的检测。

3) 通过方差滤波器得到的图像区域会在这部分被进一步判断。集成分类器由 n 个基本的分类器构成，每一个分类器 i 负责区域内的像素比较，并产生相应的后验概率 $P_i\left(y\,|\,x\right)$，其中 $y\in\{0,1\}$，然后集成分类器对这些后验概率求均值。如果该图像区域的均值大于50%，则认为该区域存在目标物体。

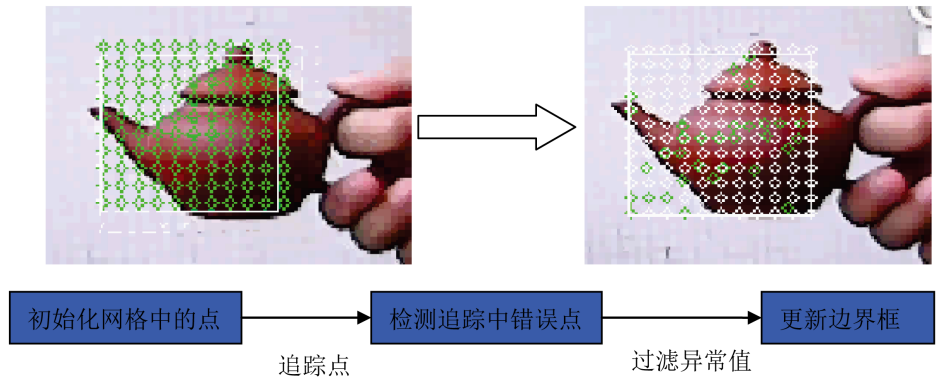


Figure 3. Tracking point selection
图 3. 选择追踪点

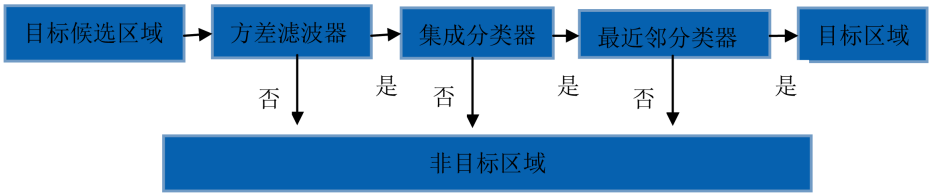


Figure 4. Tracking point selection
图 4. 选择追踪点

其原理是随机蕨分类器，初始化随机蕨分类器，遍历扫描整个图像，然后对样本进行分类，用 $c_i, i=1,2,\cdots,L$ 表示，设 $f_j, j=1,2,\cdots,N$ 为输入样本的二元特征集，则样本的所述类别如(2.1)式：

$$\hat{c}_i = \arg_{c_i} \max P\left(C=c_i \mid f_i, f_2, \cdots, f_N\right) \tag{2.1}$$

式中，C 表示类的随机变量，则有

$$P\left(C=c_i \mid f_i, f_2, \cdots, f_N\right) = \frac{P\left(f_i, f_2, \cdots, f_N \mid C=c_i\right) P\left(C=c_i\right)}{P\left(f_i, f_2, \cdots, f_N\right)} \tag{2.2}$$

设先验概率 $P(C)$ 为均匀分布，(2.2)式中分母部分与类别无关，则将(2.1)化简

$$\hat{c}_i = \arg_{c_i} \max P\left(f_i, f_2, \cdots, f_N \mid C=c_i\right) \tag{2.3}$$

二元特征 f_j 的值取决于样本中在分类器训练时随机生成的两像素位置 d_{j1} 和 d_{j2} 的灰度大小 Id_{j1}，Id_{j2} 比较的结果，即

$$f_j \begin{cases} 1 & Id_{j1} < Id_{j2} \\ 0 & \text{其他情况} \end{cases} \tag{2.4}$$

除了需要大量的特征来保证分类结果外，为了保证 f_j 之间具有足够的相关性并减少存储，假设不同组的二元特征之间相互独立，组内二元特征之间具有相关性，则(2.3)式的条件概率近似为

$$P\left(f_i, f_2, \cdots, f_N \mid C=c_i\right) = \prod_{m=1}^{M} P\left(F_m \mid C=c_i\right) \tag{2.5}$$

式中 $F_m = \left[f(m,1), f(m,2), \cdots, f(m,s) \right], m=1,2,\cdots,M$ 表示第 m 个蕨，(m,j)表示范围为$1,\cdots,N$ 的随机函数。

如从图 5 茶壶中任意选取两点 X 和 Y，比较两者亮度值，若 X 亮度大于 Y 则特征值为 1，反之为 0，每选取新的一对就是一个新的特征值，蕨的每个节点就是对每一对像素点进行比较。同一类的样本经过同个蕨就可以得出该类的结果，不同类的样本经过同个蕨则得到不同的先验概率分布。在对分类器进行训练后，利用(2.2)式就可以求出新样本的分类。

4) 最近邻分类器的主要思想是将要测试的记录与训练集的每条记录计算距离，然后选择距离最小的 K 个，将 K 个记录中的类标号的多数赋给该测试记录，如果所有的类标号一样多，则随机选择一个类标号。

利用最近邻分类器的思想，在通过集成分类器判断目标存在的区域中，取图像元 p_i 和 p_j 的相似度，N 为相关系数，S 取值范围为[0,1]之间

Figure 5. Random selection of ferns
图 5. 随机蕨取点

$$S\left(p_i,p_j\right)=\frac{N\left(p_i,p_j\right)+1}{2} \tag{2.6}$$

分别求正负最近邻相似度

$$S^{+}\left(p,M\right)=\max_{p_i^{+}\in M} S\left(p,p_i^{+}\right) \tag{2.7}$$

$$S^{-}\left(p,M\right)=\max_{p_i^{-}\in M} S\left(p,p_i^{-}\right) \tag{2.8}$$

如果一个图像区域的相对相似度 $Sr\left(p,M\right)>\theta_{NN}$，其中 $\theta_{NN}=0.6$，则图像区域认为存在目标物体，并将其加入到最近邻分类器的模板中。值越大则代表相似度越高。

$$S^{r}=\frac{s^{+}}{s^{+}+s^{-}} \tag{2.9}$$

若最近邻分类器中的模板数量超过了一定的阈值，则选择随机舍弃一些模板。

2.1.3. 学习模块

学习模块所采用的机器学习方法是 P-N 学习，即通过 P-N 学习机制来更新随机蕨分类器。作为一种半监督的机器学习算法，它针对识别器样本分类时产生的漏检和误检两种错误进行纠正。P-N 学习的流程如图 6 所示。

首先要产生样本则需要用不同尺寸的扫描窗对图像进行逐行扫描，每在一个位置就形成一个包围框，包围框所确定的图像区域称为一个图像元，图像元进入机器学习的样本集就成为一个样本。扫描产生的样本是未标签样本，需要用分类器来分类，确定它的标记。

P-N 约束对未标记的样本进行标记分类，之后训练分类器。正约束是指将未知样本标记为正样本的标记条件，这里将靠近轨迹附近的样本标记为正样本。负约束是指将未知样本标记为正样本的标记条件，这里将远离轨迹附近的样本标记为负样本。通过正负约束的相互作用来提高样本标记的准确性。

设 x 为特征空间 X 中的一个样本，y 表示为对应标记空间 $Y=\{-1,1\}$中的一个标记，那么可以使用集合空$\{X,Y\}$来表示样本空间和对应的标记。P-N 学习根据已标记的样本集合$\{X1,Y1\}$来建立分类器，并训练样本，使用没有标记过的数据 X_u 来引导分类器工作，提高分类器性能。

P-N 学习的主要步骤如下：

1) 对已有的标记样本训练出一个初始的分类器。

2) 利用分类器对未标记样本进行分类判断。

3) 识别出分类判断结果与结构约束不相符的样本，并重新对这些样本进行标记。

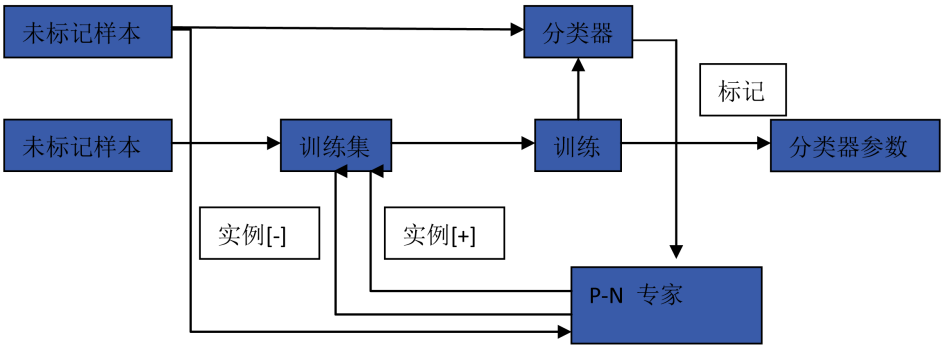


Figure 6. Learning procedure of P-N method
图 6. P-N 学习流程图

4) 将重新标记后的样本加入到训练集中，并再次训练分类器。

5) 跳转至步骤 1)循环执行。

至此，识别结果已经可以基本正确的显示，当通过语音识别获取校验信息时，根据校验信息重新训练得到新的分类器。

2.2. 语音识别

由于语音识别模块采用的是 Speech SDK 的框架，其已经提供了各种接口来实现不同的语音功能，包括语音识别引擎、语音识别上下文、语法规则、识别结果及语音合成等等[13]。针对本系统本文着重处理语音校正部分。

2.2.1. 语音识别特征提取

由于输入的信号是时域信号，时域特征不够明显，需要转化为频域信号。语音识别是为了实现单词的识别效果，进而实现语句的识别效果，需要把时域信号分割成时间片进行分析才能达到比较精确的识别效果。即，每个时间片的时域信号都进行傅里叶变换转化为频域信号。另外，由于人耳对声音频率变化的感知不是线性的，而是基于以下函数模型：

$$z = 1127 \times \log\left(1 + \frac{f}{700}\right) \tag{2.10}$$

2.2.2. 语音识别训练集和测试集

训练集是用来训练的语音模板，训练的结果是形成一个识别模型。测试集是用来测试模型准确度的语音模板。测试集所占比例通常大于训练集才能保证识别模型足够准确。判断识别模型是否准确的方法之一是，多次随机生成相同比例的测试集和训练集，求识别错误率的平均值。对多个测试集和训练集的比例进行测试。不过，为了避免出现过分类，我们选取测试集：训练集为 1:1。

2.2.3. 语音识别系统模板

针对本项目，系统只需要识别“识别结果正确”和“识别结果错误”两种结果。由于考虑到不同的人语言的频率、音调不同，我们对每一种分支设定多个模板，然后再把这些模板集合成一个模板。上文所说的测试集和训练集中的任意一条模板都是集合成的模板。首先，确定模板被分成几个 state。我们的语音长度足够大，模板被分成 18 个，并将训练集模板平均分成段，求这些模板的所有段的平均值形成平均模板。然后，对每一个原始的训练集模板，比较原始训练集模板和平均值模板，重新确定训练集模板中 feature 的 state，这样就重新调整了训练集模板的 states。最后再求所有调整过的训练接模板的所有 states 的平均值获得识别系统的识别模板。

2.2.4. 语音识别系统识别算法

识别系统采用 DP 的方式实现。这种识别系统是基于模板距离比较来分离出的分类器。这种距离计算依赖于单高斯 HMM 模型。假设模板被分成多个段，每个段被赋予一个 state，某个段的每个 feature 属于该 state 的概率服从单高斯分布。假设这个概率的值为 p，那么这个 feature 跳转到下一个 state 的概率为 1-p 即转移概率。对该值进行 log 计算并取负值结果为 edge cost。另外还有一个值为 node cost。node cost 是指输入测试模板的 feature 和识别系统模板 state 比较计算出来的距离。这个距离是高斯距离，即，考虑到 state 的方差。

3. 系统运行实验及分析

该系统的实现采用的是开源的跨平台计算机视觉库 OpenCV 以及基于 COM 视窗操作系统的

Microsoft Speech API 开发包。

3.1. 物体识别

首先针对物体识别的功能进行分析。如图 7 是系统的初始界面，共有三种读入数据的方式：摄像头读入，视频读入以及图片读入。摄像头读入是通过摄像头选定目标物体后跟踪和捕捉目标物体；视频读入则是在读入一个视频数据选定目标物体后对视频中的目标进行跟踪和捕捉目标物体；图片读入则是初始读入图片数据并选定目标物体，然后从摄像头中找到指定的目标物体。这三部分将通过包括平面移动、平面转动、复杂背景、慢速及快速运动、形变、光照、运动模糊等多面来测试系统对不同条件情况下的处理能力。

3.1.1. 摄像头读入

摄像头读入作为最核心的部分本文采用了室内室外两块具有明显光照区别的实验场景进行实验从而可以检测物体自动识别在各种情况下的效果。

首先是室内采用的是静态的物体识别测试，从最简单的方式开始检测物体识别的效果，包括摄像头的移动，遮挡，消失重现，摇晃四个方面来看。

如图 8 所列的(1) (2) (3) (4) (5) (6) (7) (8)八幅图分别是对摄像头读入的第 11、34、45、54、67、80、89、95 帧的追踪和识别结果。

1) 图 8 (1)是初始识别状态；

2) 图 8 (2)中看出当目标物体被遮挡时是肯定识别不到任何物体的，但从图(3)中可以看出物体重新被识别了；

3) 图 8 (3)中目标物体和图(1)一样得到了重新得到了识别并更新了图像元；

4) 将摄像头如图 8 (4)中向右边移动时图像元中的部分的分类器会变成变色，因为距离较远其实是有部分缺失才会导致这种情况，但整体的图像元依然能够清晰的识别，P-N 学习的方法使得部分未缺失的图像元得到继续的学习和定位；

5) 图 8 (5)将摄像头转移让目标物体消失后来测试训练集的效果；

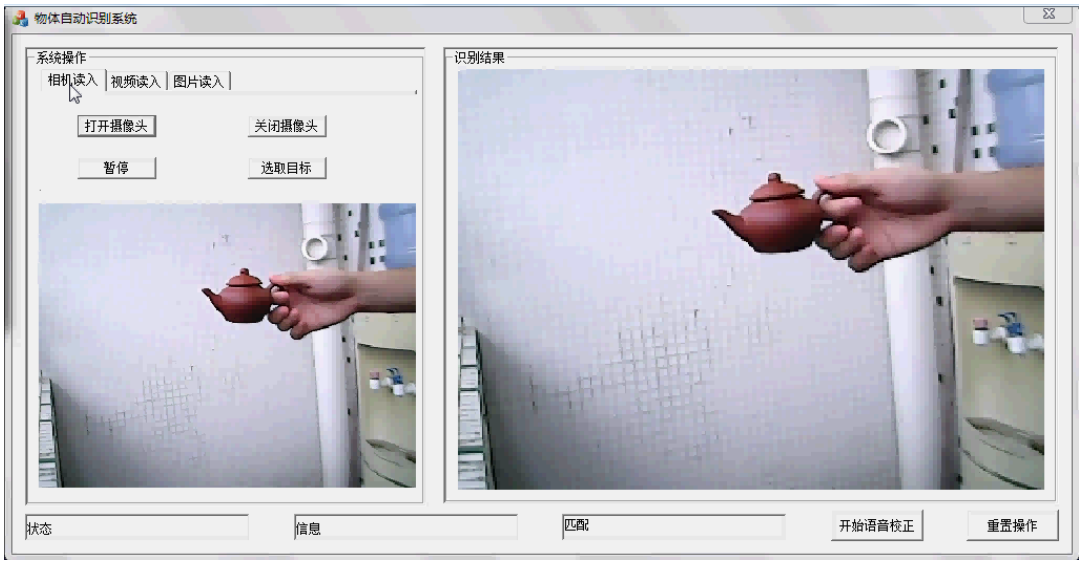


Figure 7. System demonstration
图 7. 系统界面

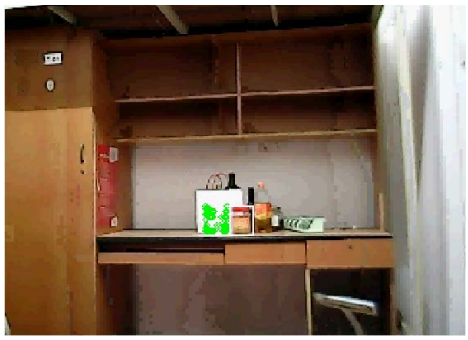
(1) Initial object
(1) 初始目标物体

(2) Object interrupt
(2) 目标物体遮挡

(3) Re-recognition of objects
(3) 目标物体重新识别

(4) Camera moves to the right
(4) 摄像头平面右移

(5) Object disappears
(5) 目标物体消失



(6) Object re-appears
(6) 目标物体再现

(7) Camera shakes
(7) 摄像头摇晃

(8) Camera shakes
(8) 摄像头摇晃

Figure 8. Test of in-door camera movement with static object
图 8. 室内摄像头移动目标物体静止测试

6) 图 8 (6)当摄像头中的图像元中重新出现有训练集的数据时，该图像就会被重新识别；

7) 图 8 (7)同时尝试将摄像头晃动，看下不清晰的图像元对识别效果有什么比较大的影响；

8) 图 8 (8)虽然图像会像图(7)中不稳定导致下半部分空白，但当摄像头得到稳定的图像后识别率依然很高；

由于室内的灯光比较昏暗，距离也比较远，从实验开始到结束目标物体的图像元基本都会有点残缺。当将实验场景移到户外光亮的地方时，该系统的识别效果就显露了出来。

如图 9 所列的(1) (2) (3) (4) (5) (6) (7) (8)八幅图分别是对摄像头读入的第 20、34、42、61、72、83、98、109 帧的追踪和识别结果。

1) 图 9 (1)是初始识别状态；

2) 从图 9 (2)中看出当目标物体移近放大时图像元的效果很好，与初始状态是几乎相同的，说明放大时并不会影响物体的识别效果；

3) 在图 9 (3)中将目标物体平面缓慢移至左上角时图像元的效果也是很好，说明在低速运动下该算法的识别效果很不错；

4) 而当目标物体像图 9 (4)中向右边突然快速移动时图像元中的部分分类器会有所缺失，但是整体的图像元依然能跟着目标物体移动，说明当目标物体消失的时候识别器就会凭着对过去目标物体识别的学习重新对其进行定位。这里面的学习方法便是 P-N 学习；

5) 同时再将目标物体再次慢速下移，如图 9 (5)效果已经远不如当初图 9 (3)能让所有识别器都能识别到目标物体的特征。但依然能凭借过去的 P-N 学习继续对其定位；

6) 让目标物体突然消失，然后过 3 s 后再次出现在摄像头范围内。图 9 (6)中可以发现目标物体的图像元的识别器已经和当初的识别效果相同了，也是因为有了 P-N 学习才能再次训练分类器；

7) 将茶壶的盖子移开后，整个茶壶的形状实质上是发生了改变，如图 9 (7)。虽然会缺少部分图像元但学习模块会根据跟踪器的跟踪结果对识别器的两个错误进行评估，并根据评估的结果生成训练样本对识别器的目标模型进行更新，同时对跟踪器的关键特征点进行更新，所以只要茶壶的主要特征点还在，虽然缺少了一个盖子但是主体依然能作为目标来继续识别；

8) 图 9 (8)中的转动其实也包含有目标物体的形变，但只要变动不是特别大能识别出主体，则目标物体可以继续被识别和跟踪。

从室内和室外两个不同的测试可以看出基于TLD算法的物体识别系统对于光照的变化是比较敏感的，一旦发生了较大的光照变化，尤其是局部目标表示的方法时，目标物体容易丢失；在光照较强时目标物体更易于识别；而当光照较弱后，目标物体的分类器丢失严重。

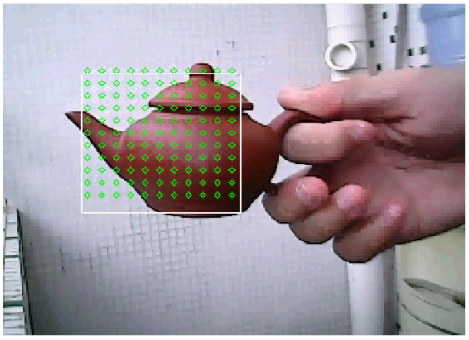
同时根据图 10 物体识别相似度可以看出物体由于在识别过程中有移动、放大、变形、消失再出现等动作，所以相似度会在某段帧间有较大的波动。其中在后期出现变形和旋转时相似度虽然只有 0.8 左右，但依然能保持很好的跟踪识别效果。所以总体的相似度还是能与初始选定的目标物体相匹配。

同时图 11 的识别率变化则反映出虽然对目标物体的相似度不能达到非常好的效果，但依然能保持比较高且稳定的识别，从而保证目标物体不会丢失。

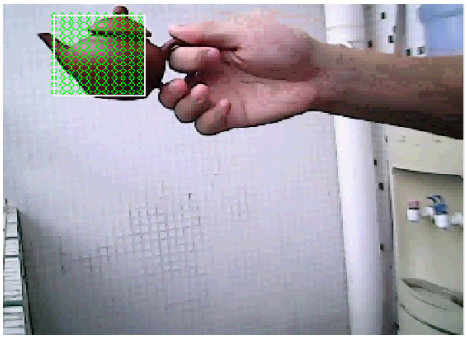
3.1.2. 视频读入

仅仅在通过摄像头在现实生活中的测试是远远不够的，因为有很多情况都容易忽视或实现，这时候就可以从 visual Tracker Benchmark 中下载大量的数据集来测试，如图 12 所示。这些数据集都是从 IEEE 等多个议会的多篇近期论文中选择出来的。这些测试数据集都尽可能的考虑测试视频所应该面对的恶劣场景的所有情况。

(1) Initial object
(1) 初始目标物体



(2) Object magnifies
(2) 目标物体放大



(3) Object moves left slowly
(3) 目标物体平面慢速左移

(4) Object moves right quickly
(4) 目标物体快速右移

(5) Object moves down slowly
(5) 目标物体平面慢速下移

(6) Object disappears and re-appears
(6) 目标物体右消失后再现

(7) Deformation of object
(7) 目标物体变形

(8) Rotation of object
(8) 目标物体转动

Figure 9. Test of out-door camera movement with static object
图 9. 室外目标物体在摄像头中运动测试

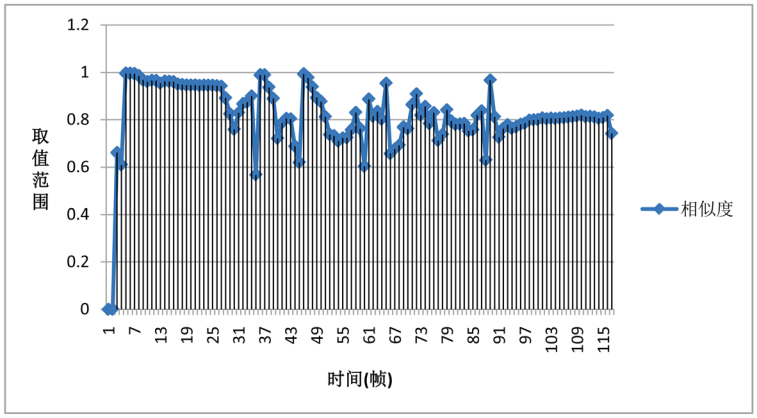


Figure 10. The line chart of similarity rate on object recognition
图 10. 物体识别相似度变化折线图

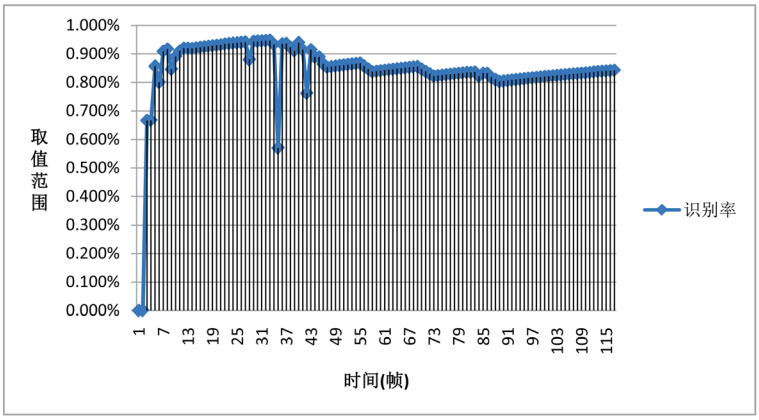


Figure 11. The line chart of object recognition rate
图 11. 物体识别识别率变化折线图

TB-50 Sequences.

Figure 12. The test data set
图 12. 测试数据集[5]

我们从所有数据集中选取了几类不同环境下的数据集作为测试内容。其中以 David 最为经典，因为它所包含的测试有光照变化(IV)、尺度变化(SV)、闭塞(OCC)、变形(DEF)、运动模糊(MB)、内平面旋转(IPR)及外平面旋转(OPR)。所以将其作为接下来的测试内容。

如图 13 所列的(1) (2) (3) (4) (5) (6)六幅图分别是对视频读入的第 12、14、42、61、72、83、98、109 帧的追踪和识别结果。

1) 图(1)是初始读入的状态；

2) 从图(2)中看出当目标物体读入的时候环境已经是非常昏暗的，只能大概看的出是人脸的轮廓，但依然能识别出目标物体的位置；

3) 图(3)中目标物体的头发生了扭动，但依然不会影响识别的准确度；

4) 图(4) (5)中目标物体的亮度慢慢发生了变化，因此识别的效果更好，图像元所要跟踪的范围也可以缩得更小；

5) 根据图(6)当目标物体脱下眼镜时属于面部发生了形变的变化，但其主要的脸部特征并没有发生改变，所要依然能准确识别出目标物体的脸部。

从图 13 中可以看出视频的测试也是比较准确的，由于物体识别的相似度和识别率效果都与前者相同则之后都不再给出识别过程中的数据信息。

3.1.3. 图片读入

图片读入主要是先对一张图片中的指定物体进行选定，然后将指定物体作为目标物体通过摄像头来识别和跟踪。如图 14 中左下边是读入的图片内容，在图片内选定酒品作为识别物体，而右边则会根据选定的物体进行识别，最后识别的效果也和之前的识别效果相类似。

3.2. 语音识别

正如前面的工作内容所描述，由于读入的语音内容少，只需要对“识别结果正确”和“识别结果错误”两个语音进行处理，所以重点是更好的调试和实现它。所以实验部分也会比较简单，当识别到目标物体时只要念到“识别结果正确”就会弹出“识别结果正确”的提示框，然后可以继续识别；当无法识别或识别不到目标物体的时候只要念“识别结果错误”就会弹出“识别结果错误”的提示框，然后可以重新圈定目标物体并重新识别。图 15 中的六幅图是之前所有实验结果的识别情况：

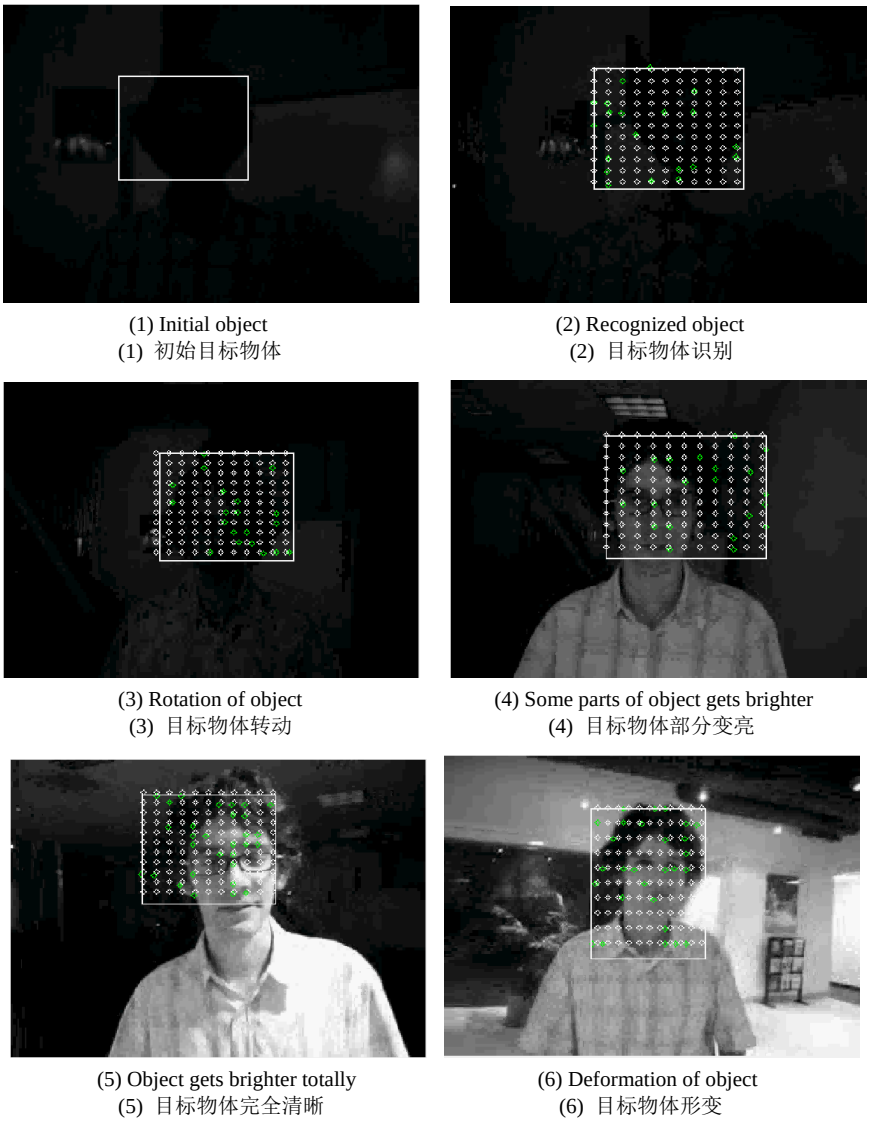
4. 物体自动识别中的难点与展望

本文在对物体识别的测试过程后提出了许多的问题和想法，里面包含了对该系统当下面临的难点和展望做出了分析。

其中物体识别部分大概可以分为两类，一是基于背景建模，该种方法主要是利用背景建模，提取出前景运动的目标，在目标区域内进行特征提取，然后利用分类器进行分类看是不是需要识别的物体。虽然这并不是本文所采用的方法但仍具有值得借鉴的意义。二是基于统计学习的方法，该方法是目前比较常用的方法，因为根据大量的样本学习所获得的样本数据比但从图像背景中获取的数据要更加准确，它所提取的特征主要有目标物体的灰度、边缘、纹理、颜色、梯度直方图等信息。但同时这些特征也会为分类器带来很多的麻烦，也就导致物体识别中出现许多的难点：

1) 物体的形状各不相同，复杂的背景加上不同的光照环境，若没有提前的指定和预处理的数据很难做到真正的精确；

2) 现在所指的物体大多是固态的，当有其他形态的问题需要识别时所提取的特征在特征空间中的分布不够紧凑；



(1) Initial object
(1) 初始目标物体

(2) Recognized object
(2) 目标物体识别

(3) Rotation of object
(3) 目标物体转动

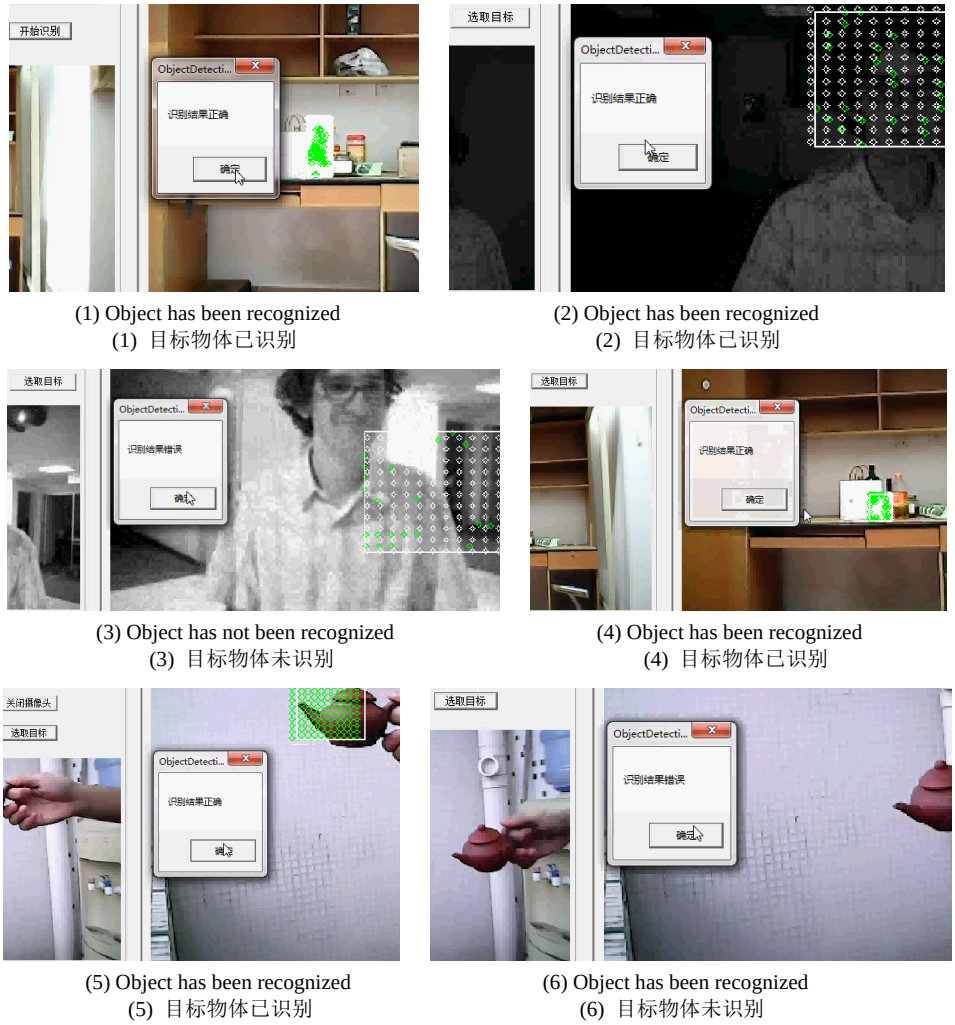
(4) Some parts of object gets brighter
(4) 目标物体部分变亮

(5) Object gets brighter totally
(5) 目标物体完全清晰

(6) Deformation of object
(6) 目标物体形变

Figure 13. Test of object movement in video
图 13. 目标物体视频中运动测试

Figure 14. Object recognition with image
图 14. 目标物体利用图像识别



(1) Object has been recognized
(1) 目标物体已识别

(2) Object has been recognized
(2) 目标物体已识别

(3) Object has not been recognized
(3) 目标物体未识别

(4) Object has been recognized
(4) 目标物体已识别

(5) Object has been recognized
(5) 目标物体已识别

(6) Object has been recognized
(6) 目标物体未识别

Figure 15. Voice implementation effect in the process of object recognition
图 15. 目标物体识别过程中的语音实现效果

3) 因为以特征为主要的样本，分类器的性能自然也更加容易受到训练样本的影响；

4) 离线训练时的负样本无法涵盖所有真实应用场景的情况而导致需要更多的硬件设备来支持。

但同时可以发现，随着算法的不断优化和改进，物体识别技术的准确率和速度也不断的提高，虽然距离真正的实用性还有很远的路要走，但在未来的物体识别技术的研究能有很多提高的地方[14]：

1) 要不断改进算法就需要对分类器中的各种数据进行分析，收集更多的有效的训练样本很有必要，这些样本需要对应于通用场景的场合、某一固定场景但有目标物体正样本和负样本的场合以及高像素度的扩充场合；

2) 单目视觉已经不足已解决现实中的复杂情况，可以采用多摄像头或利用深度信息来识别物体。同时再多摄像头下物体的形状更容易被描述，可以更加深入的探讨多目视觉中的物体识别技术；

3) 虽然实验也考虑并采用了许多恶劣的环境，但系统还需要对部分遮挡、分辨率低、远距离的、携带大面积物件的行人进行更加准确的识别并保持低误报率才能达到实际应用的效果。同时可以建立专门针对遮挡、低分辨率和远距离的行人测试数据库来进一步完善；

4) 目前大部分的检测器都是通用检测器，但缺乏在特殊场景下尤其是摄像头静止不动的监控场合如

何利用增量学习、在线学习等算法将通用的检测器迁移到特殊场景中，使物体识别技术能在识别的过程中通过学习提高性能。

而语音部分虽然在现在的系统中只需要识别简短的两句话，但所采用的语音系统比较落后，且缺乏自主语音的设计，以后还需要在这一方面有更多的扩展才能让整个系统更加的完善。因而在针对需要更多的语音进行处理时可以利用云计算中的虚拟化技术和并行计算技术以及更加规范化和形式化的云端应用服务从而减少语音识别计算的约束，实现更加完善的语音功能。

5. 结论

本文的物体自动识别系统不仅是对物体跟踪识别算法的实现，更拥有着当前市面不多的语音交互功能。该系统既能作为金融公司、重要企业进行身份认证与识别应用的原型，同时也能为智慧城市的构建提供智能监控功能，更能成为当今车载智能领域阔进的一个重要部分，会有很好的社会应用前景。

在未来，我们致力于两方面的工作：一是不断改进和学习更多更优秀的物体识别和算法，同时将界面优化，增加更多的 API 接口使其能够在各种硬件上得到兼容；二是逐渐摆脱已有的语音 SDK 包，与更多语音技术的人合作利用更优秀的技术来完善和增加更好的语音功能。

参考文献 (References)

[1] Koller, D., Weber, J. and Malik, J. (1994) Robust Multiple Car Tracking with Occlusion Reasoning. *Proceedings of* 3*rd European Conference on Computer Vision* (*ECCV*'94), **800**, 189-196. http://dx.doi.org/10.1007/3-540-57956-7_22

[2] Gori, F., Santarsiero, M., Piquero, G., Mondello, A. and Simon, R. (2001) Partially Polarized Gaussian Schell-Model Beams. *Journal of Optics*: *A Pure and Applied Optics*, **3**, 1-9. http://dx.doi.org/10.1088/1464-4258/3/1/301

[3] Comaniciu, D., Ramesh, V. and Meer, P. (2003) Kernel-Based Object Tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence, **25**, 564-577.

[4] Barron, J., *et al.* (1992) Performance of Optical Flow Techniques. *Proceedings of the International Conference on Computer Vision & Pattern Recognition*, Champaign, 15-18 June 1992, 236-242. http://dx.doi.org/10.1109/cvpr.1992.223269

[5] VTB (2013) Visual Tracker Benchmark. http://www.visual-tracking.net

[6] VOT (2013) Visual Object Tracking. http://www.votchallenge.net

[7] Wu, Yi, *et al.* (2013) Online Object Tracking: A Benchmark. *Proceedings/CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, **9**, 2411-2418. http://dx.doi.org/10.1109/cvpr.2013.312

[8] Posdamer, J.L., *et al.* (1981) Computer Geometric Modeling for Machine Perception of Three-Dimensional Solids. *Technical Symposium East. International Society for Optics and Photonics*, 29 October 1981.

[9] Engel, F.L. (1977) Visual Conspicuity, Visual Search and Fixation Tendencies of the Eye. *Vision Research*, **17**, 95-108. http://dx.doi.org/10.1016/0042-6989(77)90207-3

[10] Collins, R., Lipton, A., Fujiyoshi, H. and Kanade, T. (2001) Algorithms for Cooperative Multisensor Surveillance. *Proceedings of the IEEE*, **89**, 1456-1477. http://dx.doi.org/10.1109/5.959341

[11] Kalal, Z., Mikolajczyk, K. and Matas, J. (2012) Tracking-Learning-Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **34**, 1409-1422. http://dx.doi.org/10.1109/TPAMI.2011.239

[12] Bradski, G.R., *et al.* (2014) Learning Open CV. Oreilly Media.

[13] Luo, J.W. (2009) Program Design and Implementation of Voice Based on Microsoft Speech SDK. *Bulletin of Advanced Technology Research*, **3**, 22-25.

[14] Su, S.Z., Li, S.Z., Chen, S.Y., Cai, G.R. and Wu, Y.D. (2012) Pedestrian Detection Technology Reviewed. *Acta Electronica Sinica*, **40**, 814-820.