

****

**J I A N G S U U N I V E R S I T Y**

**本 科 毕 业 论 文**

基于Java的视频动作识别系统

**Java - based Video Action Recognition System**

学 院 名 称： 计算机科学与通信工程学院

专 业 班 级： 计算机1302班

学 生 姓 名： 曾 彪

指导教师姓名： 詹永照

指导教师职称： 教 授

2017 年 5 月

基于Java的视频动作识别系统

专业班级：计算机1302班 学生姓名：曾彪

指导教师：詹永照 职称：教授

**摘要** 视频动作识别是目前计算机视觉领域发展的一个热点方向，在计算机视觉领域，很多技术都已经逐渐趋于成熟，比如说人脸识别，早在前几年各大公司就相继宣布人脸识别率达到99.9以上，标志着机器学习的应用已经不再是遥不可及的童话，与此同时，随着技术的发展和社会生活的需要，越来越多的学者开始转向于研究它的另一个分支——视频动作识别。

视频动作识别的基本原理和人脸识别一样，但由于人体动作的特征性，整个动作是由许多帧连贯而成的，每帧之间是上下文相关的，采取的特征提取方法等也就和人脸识别有所不同。

由于以往的论文基本着重于理论的介绍，很多初学者看完后往往是晕头转向，更多的细节实现更是一带而过，对于机器学习这一庞大的知识体系，很容易让人抓不到重点，以至于让后面的研究者花费大量的时间在环境以及平台兼容性上，而不是专心研究算法。基于各方因素，本文的主要研究内容主要分为以下几点：

1.采用了目前世界上比较流行，但在机器视频领域选用的较少的Java语言作为开发平台，并详细介绍了系统开发的结构和思路。

2.设计了一个协同训练算法，对两个分类器进行协同训练集成融合，选取置信度较高的视频数据作为训练样本进行迭代训练，以解决标记数据不容易获取、标记代价大等问题。

3.由于在视频里一个动作往往是可以分解成很多个小的动作的，以一定的时间间隔作为分割，每个时间范围内的视频序列往往已经足够用来表示一个动作，因此我们以此作为依据，一个视频就可以划分成若干个动作的集合，这样也可以用来解决视频标本数据不足的问题。

4.原理同上一点一样，既然训练的时候一个视频可以提取出若干个动作特征，那识别的时候同样也可以对一个待识别的视频数据提取，然后对每个动作特征分别进行识别，最后采用一定的投票算法选取出一个可能性较高的结果。

。

**关键词：**动作识别 机器学习 行为识别 机器视觉 Java

**Spark Model Optimization For Large Data Applications**

**Abstract** No

d.

**Key words** Big Data Machine Learning RNNLM Spark

# 第一章 绪论

## 1.1 课题背景及意义

视频动作识别是机器学习的一个分支，本质上都是运用大量的有标记的数据去训练分类器，然后再用训练出来的分类器来进行样本的识别，但是与人脸识别不同的是，和目前比较热门的人脸识别相比，无论是国内还是国外，视频动作识别这方面的研究都还比较少，难度也相对较高，而且目前比较流行的平台大多是基于Python、matlab、C++的，基于java的则更是凤毛麟角了，人脸识别所需要的输入数据只是静态的图片，只要提取出当前图片的特征，再进行训练和比较，就很容易了。虽然也是视频，我们经常见到的通过摄像头扫瞄人脸，然后进行实时跟踪的例子也基本上是属于人脸识别的范畴，因为它相当于把视频拆分成一帧一帧的图片，然后只要在每一帧中分别识别就可以了。而视频动作识别则不仅仅需要识别出每一帧图片，而且每帧图片之间还是上下文相关的，它们之间存在着某种关联，只有特定的组合才能成为一个动作。另一方面，目前人工智能领域比较热门的动作识别方法主要是通过大量的有标记数据样本来进行训练，这样的话虽然最终的分类器的精度比较高，但是有标记的样本往往需要具体样本所属领域的专家进行手工标记，这样一来要获取大量有标记的样本，所要耗费的人力财力极高。

在人们的日常生活中，信息可以说是无处不在，各种各样的信息充斥着我们的社交媒体，一方面，信息时代给我们的生产生活带来了极大的便利，手指随便一动，各种我们想要的信息就出现在屏幕上，另一方面，信息的增长速度也逐渐超出了人们的控制，各种我们想要的不想要的信息，都展现在我们面前，尤其是一些虚假信息的产生，不仅浪费了人们大把大把的时间，还会对人的判断产生误导。如果是文字的信息还好，基本上随便一瞥，大概的信息便了然于心，即便是交给计算机云处理，也极为的方便，只需要设置一些关键字，便可以很便捷地过滤出我们想要的信息。但是对于视频信息，情况则又不同了，互联网上视频信息也是成千上万，鉴于视频的特征性，如果想要知道视频里有什么信息的话则需要把视频完整的播放一遍，这样一来，如果视频很长的话，浪费的时间可是不可忽视的。

基于此，一种能够自动识别出视频里面动作的系统就应运而生，我们可以把视频动作识别这种特别耗时的操作交给计算机去处理。计算机系统首先从一大堆有标记的视频数据中提取出相关视频中人体动作的特征，然后根据分类器的训练方法用这些特征去训练出一个强大的分类器，后面的事情就方便了，当我们需要知道一个视频里面的动作类别时，就不用人工地把整个视频全看完，直接把视频提交给计算机，计算机从视频中提取出动作特征，运用分类器进行识别，把视频的类别告诉用户，当视频非常多时，也可以交给计算机去批量处理，从而省下了大量宝贵的时间。

应用人工智能使计算机也像人类一样拥有“视觉”，从计算机的角度去分析视频，把所需要的信息展现在我们面前，初期的目标是设计一个基于java的软件，用户通过手动导入相关视频，软件系统使用本课题中的相关方法进行分析，给出结果，使计算机能够帮我们人类快速识别出视频所包含的信息，并把结果反馈给用户以供决策，尽管这些步骤看上去非常粗糙、繁琐，但是我相信在不久的将来，我们甚至可以进一步地把这一成果应用到互联网领域，到那时候互联网上的每台计算机可以对识别出的视频贴上“标签”，而用户可以预先设置好想要看到的标签和不想看到的标签，计算机可以根据用户的需求进行推送和屏蔽，完全不需要人工的干预。

当技术发展到一定阶段时，还可以往更深的方向探索。比如我们可以建立一个庞大的服务器，对互联网上的视频进行批量的识别，并贴上视频的类别标签，方便用户进行查找和鉴别，当好友向你发送一个视频时，系统会先进行扫描，把视频的内容信息通过简要的文字描述给你，你再选择查看或者忽略。

## 1.2 研究现状

和目前比较热门的人脸识别相比，无论是国内还是国外，视频动作识别这方面的研究都还比较少，难度也相对较高，而且目前比较流行的平台大多是基于Python、matlab、C++的，基于java的则更是凤毛麟角了，人脸识别所需要的输入数据只是静态的图片，只要提取出当前图片的特征，再进行训练和比较，就很容易了。虽然也是视频，我们经常见到的通过摄像头扫瞄人脸，然后进行实时跟踪的例子也基本上是属于人脸识别的范畴，因为它相当于把视频拆分成一帧一帧的图片，然后只要在每一帧中分别识别就可以了。而视频动作识别则不仅仅需要识别出每一帧图片，而且每帧图片之间还是上下文相关的，它们之间存在着某种关联，只有特定的组合才能成为一个动作。另一方面，目前人工智能领域比较热门的动作识别方法主要是通过大量的有标记数据样本来进行训练，这样的话虽然最终的分类器的精度比较高，但是有标记的样本往往需要具体样本所属领域的专家进行手工标记，这样一来要获取大量有标记的样本，所要耗费的人力财力极高。

## 1.3 课题结构及研究内容

**1.4 可行性分析**

# 第二章 系统相关技术介绍

# 第三章 系统设计

## 3.1 需求分析、功能需求

本课题设计的是一个基于java的视频动作识别系统，开发平台主要是eclipse，视频动作识别系统要求是要能够对输入的视频进行相应的处理，能够判断出视频中含有的人体动作的类别，在本次研究中，我们采用的是比较经典的kthdata人体动作视频库，里面含有6种不同的人体动作，系统所要实现的是给用户提供一个友好可视的用户操作界面，在界面上通过按钮的点击完成视频的识别功能，并在具体的位置显示出处理的结果。

在传统的动作识别的研究的基础上，本课题更加注重的是系统的实现，因为以往的动作识别的研究讲的大多都是原理，是一个比较笼统的概念，至于实现的细节，往往提的很少，然而由于视频动作识别的复杂性，很多细小的部分都是不可忽视的，有的时候一个小小的参数设置的不同，都可能会导致特征提取、识别结果的截然不同。鉴于此，本次课题选择了一个比较具体的语言java，一个在动作识别领域用的很少的语言，给后面的java动作识别研究者一个铺垫，少做一些麻烦的事。

在java平台上编写视频动作识别系统还需要一个工具Opencv，这是一个

系统功能模块图、功能描述

系统流程图

用户界面设计

# 第四章 系统功能实现

## 4.1 训练数据的准备

要运行该程序，必须将OpenCV目录下build/bin目录下的opencv\_ffmpeg的dll文件（有2个，一个是32位，另一个是64位，比如我的是64位的环境，opencv2.4.9，就用了C:\mysoftware\opencv\build\x64\vc12\bin目录下的opencv\_ffmpeg249\_64.dll文件）复制到电脑上jdk的安装目录下的bin目录（以我的为例，就是C:\Program Files\Java\jdk1.8.0\_91\bin）下面才可以。这个问题真的很崩溃啊，在VisualStudio里只要配置好Opencv环境就一点问题没有了，但是按照网上的教程配置好java的Opencv的环境后，编译一点问题没有，但是死活就是识别不出视频文件，而且网上了好多文章都没有讲到还要配置这个dll文件的，对于以前从来没有接触过这方面知识的小白来说简直要抓狂了有没有，最后还是在看到国外一个有点类似问题的博客之后获得的灵感。

# 第五章 总结和展望