浙江大学

本科生毕业论文

终稿

姓名与学号 戴丹 3073031149

指导教师 熊蓉副教授

年级与专业 自动化（自动化 0705）

所在学院 控制科学与工程学系

摘要

手势识别技术，是一种高效、直接、自然的人机交互方式，属于计算机视觉领域中较为活跃的研究主题。同时，手的高自由度导致手势复杂、变化多样、移动高速等特性，使手势识别技术成为极具挑战性的课题之一。本文在研究梯度方向直方图（Histogram of Orientation Gradient，HOG）特征提取算法和支持向量机（Support Vector Machine，SVM）学习分类算法的基础上，借助 OpenCV2.0 视觉库和 Visual Studio C++实现了基于 HOG 的特征提取和多尺度下检测模块；借助 LibSVM 和 Matlab 实现了基于 SVM 的学习分类模块。在检测到手的条件下，本文借助 YCrCb 肤色模型对不同手势进行识别。实验统计，所开发算法对张开状态的手的检测，达到了 90%的命中率和 18%的误识率；在背景与肤色高差别情况下，能实现对不同手势的识别。

关键词： 梯度方向直方图；支持向量机 ；YCrCb 肤色模型； OpenCV2.0 ；LibSVM

Abstract

Gesture Recognition, which is efficient, direct and nature, has become one of the key techniques of human-compute interaction. Meanwhile, the high degree of freedom of hand results in the complexity of gesture, diversity of variation and rapidity of movement, which makes gesture recognition full of challenges. The author mainly focuses on the Histogram of Orientation Gradient(HOG) Algorithm for extracting the features and the Support Vector Machine(SVM) Algorithm for learning and classifying. Depending on the OpenCV2.0 visual library and Visual Studio, the author implements the module of extracting features and muti-sclaes detection with C++ language. Also, with the help of LibSVM software package and Matlab, the author constructs the block of learning and classifying. Then, under the condition of accurate detection, the author uses the YCrCb Skin Model to recognizing different gestures. At last, the result of study shows the 90% high hit rate and 18% relatively low false recognition rate for the open-state hand gesture. And the study achieves the goal of recognizing different gestures with the restriction of background.

Keywords: Histogram of Orientation Gradient (HOG); Support Vector Machine (SVM); YCrCb Skin Model; OpenCV2.0; LibSVM

目录

一、引言

1.1 背景介绍

随着数字信息技术的高速发展，人们对二维信息的处理，即数字图像处理的技术愈发成熟。对于计算机，图像的数学本质就是一个矩阵，其维数可以是 20*20，640*480 甚至更高。对数字图像处理的前提是建立在计算机不断发展的基础上的，如果没有计算机高效的运算能力，这一切只是水中月。总而言之，计算机技术和信息技术的日新月异为基于图像的手势识别提供了坚实可靠的技术基础。

计算机已经成为人们生活工作中不可替代的工具，于是人与计算机的交互成为人们乐于研究的一个话题。只有自然的，方便的人机交互才是人们所追求的。从最原始的 DOS 界面到如今的以鼠标为媒介的图形界面，人机交互正是在不断的友好化，人性化，简单化和自然化。而最近这些年的语音识别，人脸识别，手势识别，人体动作识别甚至脑机交互系统（Brain and Computer Interface）其目的也都是使人机交互走向更为自然简单的道路。人机交互的发展历史，应该是从人类适应计算机到计算机不断地适应人类的发展史。卡内基·梅隆大学的 Dan R.Olsen 教授说:“人机交互是未来的计算机科学。我们已经花费了至少 50 年的时间来学习如何制造计算机以及如何编写计算机程序，下一个新领域自然是让计算机服务并适应于人类的需要，而不是强迫人类去适应计算机。”追求自然和谐的生存是人类发展的理想模式，其中自然和谐的人机交互模式是以直接操纵为主的、与命令语言特别是自然语言共存的人机交互形式。理想的人机交互模式就是用户自由，直接操纵 (Direct Manipulation)的用户界面，它已成为未来的发展趋势，而作为其中重要内容的手势识别技术将变成研究热点

更进一步上升到哲学层面，对手势识别技术的研究，又何尝不是源于人类对人与机器的共生的深入探索呢！人在自然界中其生理机能绝对不属于强者，而机器可以很好弥补人类生理机能上的不足，于是人与机器的完美结合又何尝不是人类所向往的呢！也许到那时人的定义不再限于生物层面，机器已经成为“人”这一定义的无限延伸。

由于计算机硬件的飞速发展，人机界面的日臻完善和人们生活质量的提高，给基于视觉和图像的手势识别发展带来了机遇。IBM 与 Microsoft 等公司也将基于视觉的手势识别接口应用于商业领域中。

1、在虚拟现实环境中的应用，可以对环境和虚拟物体进行控制。在目标操作界面上使

用手来完成虚拟环境下的浏览，选择和操纵。利用不同定义的手势，来控制虚拟物体的前进和转弯，或者通过真实的手部运动来控制增强现实环境中的镜像手部运动。

2、智能家电、控制领域的应用。美洲虎公司推出了基于手势识别的车载汽车控制系统。另外有通过手势远程控制电视的系统，也有视频游戏的用户界面，也可以通过手势命令如“放大、全景和倾斜”控制视频摄像机。在计算机控制的设备中，手部被视作为灵活高效的控制环节，在机器人控制和远程机器人操作中得到了应用。在危险地点、太空或者特殊场合，不便直接操控，需要更自然的人机界面。

3、儿童、老人或聋哑人的教育和生活。通过人机接口，可以完成聋哑人和计算机的自然交流，提高其受教育的能力。同时可以建立起正常人和聋哑人沟通的管道，让正常人能够“听”懂聋哑人的“话”。在聋哑人教育方面也有一系列的手语识别系统问世。

1.2 研究现状

1.2.1 特征描述与提取算法

- **基于肤色模型[1]**

肤色检测主要是根据肤色在颜色空间上的分布特征来检测图像中的肌肤区域。Rossotti 与 Angelopoulou 分别在 1983 年与 1999 年的研究成果证明了在生物和物理上肤色分布的一致性，指出尽管人的肤色因人种的不同而不同，呈现出不同的颜色，但是在排除了亮度、视环境等对肤色的影响后，皮肤的色调基本一致。这为利用颜色信息进行皮肤检测的可能性提供了有利的证据。

YCbCr 色彩模型中 Y 是亮度，而 Cb 和 Cr 是色度信息。YCbCr 可由 RGB 线性变化得到。虽然 YCrCb 模型能在一定程度上削弱 Y 亮度对颜色的影响，但简单的排除 Y 的影响将会削弱模型鲁棒性。因而需要对模型进行非线性分段色彩变换。经过这样的非线性分段色彩变换，我们得到肤色在 YCrCb 空间中的分布集中在一个椭圆中。按照传统的方法，我们将用一个椭圆来近似表示这个肤色区域。这样我们可以将肤色点与非肤色点进行分离，对图像进行二值化。在椭圆内的点设置为 255，不在椭圆内的点设置为 0。

肤色模型在背景与肤色不相近的情况下有相当好的检测率。其缺陷在于对环境与光线等的依赖性。并且无法仅根据肤色模型来判断手势，因为肤色不是手所具有的独特特征。

➢ 基于 Haar_Like 特征[2,3]

这些简单的特征源自 Papageorgiou 等人提出的 Haar 基函数。更具体点，我们用的是三种特征。两个矩形区域的像素总和（eg:亮度值总和）之差称为“二矩形特征值”。同样大小的两个矩形区域有水平和垂直两种方式（图 1）。

Haar 特征是简单的，但是对于一副 20*20 的图像其特征值是巨大的，由于一副图像的特征值总数取决于特征矩形原型，特征矩形的大小，特征矩形在图像中的位置。于是引入了积分图进行优化计算。[2,3]

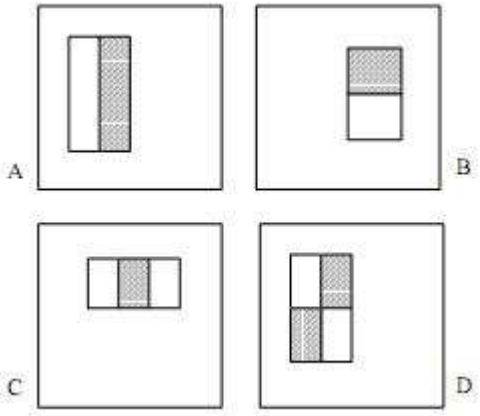


图 1.1 封闭监测区域中的矩形特征示例。灰色区域像素和与白色区域像素和之差表示矩形征值。A 和 B 表示二矩形特征，C 表示三矩形特征，D 表示四矩形特征。

利用积分图作为过渡表示，矩形特征可以很快的计算。位于 x, y 处的积分图包含 x 左边和 y 上边的像素值之和。利用积分图任何矩形区域中像素之和可利用积分图中四个参考点值来计算。Paul Viola 和 Micheal J.Jones 提出的简单 Haar 特征对于旋转很敏感，为了达到旋转不变性，Rainer Lienhart 和 Jochen Maydt 提出了扩展的 Haar 特征[4]，引入了具有 45° 旋转的 Haar 特征和中心矩形特征，并剔除了四矩形特征。

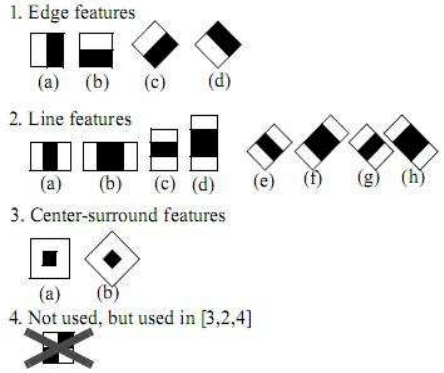


图 1.2 扩展 Haar 特征，引入了旋转 45° 和中心矩形特征进行优化

针对巨大的特征值数目和简单的特征描述，有必要采用比较强大的学习算法进行弥补，从海量的特征中选取最具代表性的特征来描述对象。相应的，Viola 提出了 AdaBoost 算法。最致命的缺点是，Haar 特征对于柔性的，变化性大的对象，譬如手的表征效果并不好。Haar 特征适用于对象中局部位置固定的情况，例如人脸中眼睛、鼻子、嘴巴的位置是相对固定的。

- **基于尺度不变（SIFT）特征[5~8]**

SIFT算法由D.G.Lowe 1999年提出[5]，2004年完善总结[6]。后来Y.Ke将其描述子部分用PCA代替直方图的方式，对其进行改进。其主要思想是一种提取局部特征的算法，在尺度空间寻找极值点，提取位置，尺度，旋转不变量。

首先需要说明何谓尺度空间。尺度空间实际是针对图像的模糊程度而言的，并不是图像的像素大小。为了生成不同尺度的图像，需要用到二维的高斯函数进行卷积模糊。接下来需要生成高斯金子塔，所谓的金字塔是针对降采样而言的。在每个维度 octave 上对图像进行尺度变换，即用具有不同σ高斯函数对图像模糊；再对每个维度最后的图像进行降采样，因而形成高斯金字塔：

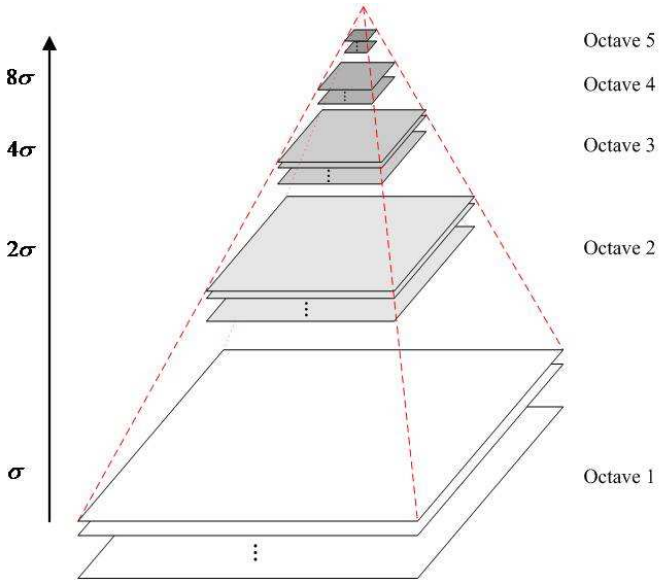


图 1.3 高斯金字塔

利用高斯金字塔生成高斯差分图像，即同一 octave 相邻两幅图像相减得到差分高斯图像 DoG（相当于对尺度维度进行微分），在 DoG 中寻找特征点，其具有尺度不变性。[8]

接下来便是在高斯差分金字塔中寻找极值点，每一个采样点要和它所有的相邻点比较，看其是否比它的图像域和尺度域的相邻点大或者小。如图所示，中间的检测点和它同尺度的8个相邻点和上下相邻尺度对应的9×2个点共26个点比较，以确保在尺度空间和二维图像空间都检测到极值点。

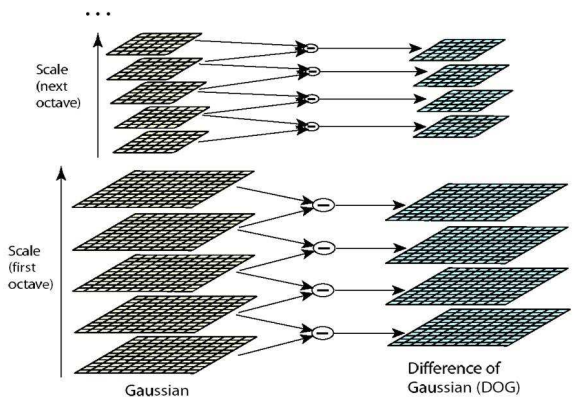


图 1.4 高斯差分图像

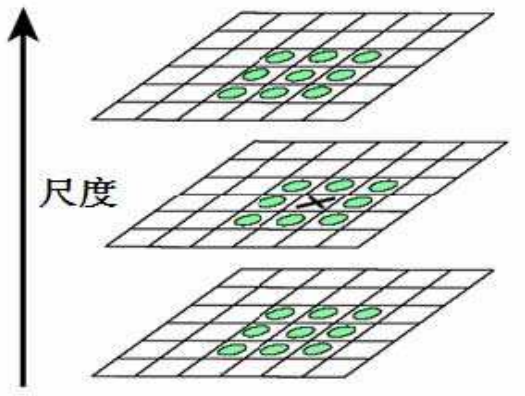


图 1.5 尺度空间寻找极值点

找到极值点后，需要精确到亚像素级别，并排除一些低对比度和高斯模糊时边缘影响的点。面阵摄像机的成像面以像素为最小单位。

通过尺度不变性求极值点，可以使其具有缩放不变的性质，利用特征点邻域像素的梯度方向分布特性，我们可以为每个特征点指定方向参数方向，从而使描述子对图像旋转具有不变性。至此我们确定了每个特征点，其信息包括：尺度空间坐标，主方向，梯度值。

接下来是对每个特征点，利用其周围点的梯度和方向信息对该点进行更详细的描述称为特征点子描述。描述的目的是在关键点计算后，用一组向量将这个关键点描述出来，这个描述子不但包括关键点，也包括关键点周围对其有贡献的像素点。用来作为目标匹配的依据，也可使关键点具有更多的不变特性，如光照变化、3D视点变化等。

可见SIFT与Haar特征比复杂很多，也具有更多对象信息。一般与比较简单的学习分类方法相结合，如支持向量机，简单神经网络，或者是决策树进行模板匹配。SIFT 算法具有以下优点：尺度不变形，旋转不变形，光照及3D视点鲁棒性。但SIFT的缺点是适用于纹理比较丰富的对象，并且会对抑制边缘特征，这对于手而言是不利的。

- **基于傅里叶描述（FD）[9]**

傅里叶描述是基于物体轮廓的描述，由Charles Zahn和Ralph Roskies最先提出，由于FD对曲线的描述具有尺度，旋转与位置的不变性，从而之后被应用于对物体的轮廓描述。尤其是对于手这种轮廓是很重要特征的对象。

假设曲线的参数表达式为： $z(l)=(x(l),y(l))$，其中 l 为弧度 $0\le l\le L$ 。

用 $\theta(l)$ 表示曲线在距起点 l 处该点的切线方向，其中 $\delta_0=\theta(0)$ 表示起点处切线方向；用 $\varphi(l)$ 表示曲线在距起点 l 处与起点处切线方向的夹角；用 $\varphi^*(t)=\varphi(\frac{Lt}{2\pi})+t$ 表示归一化的 $\varphi(l)$，则 $\frac{Lt}{2\pi}=l$ 。具体见图

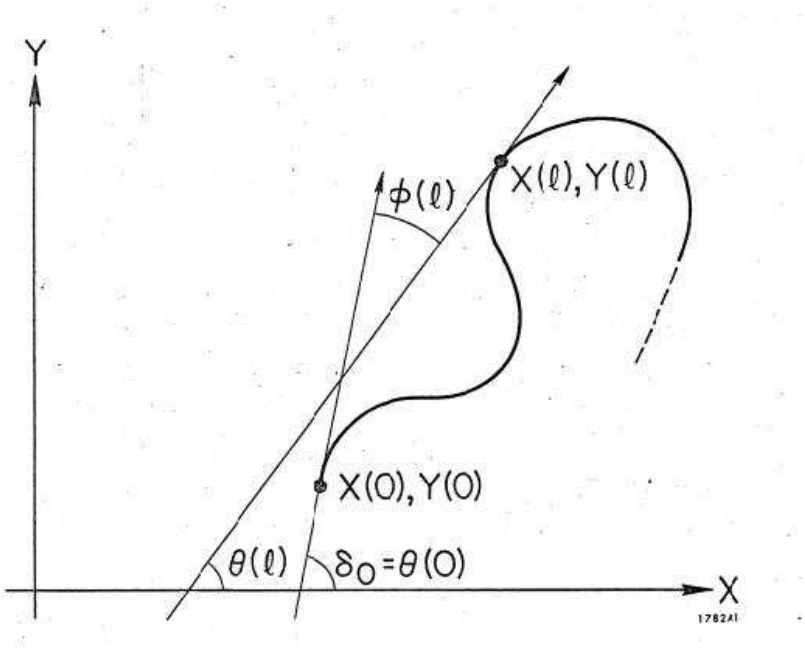


图 1.6 傅里叶描述曲线各参数方程

由 $\varphi^*(t)$ 傅里叶级数知道：可以得到曲线的幅频特性 A_k ~K 和相频特性 α_k ~K。

可以看到 z_0 决定着曲线位置，δ_0 决定着曲线旋转方向，L 决定着曲线尺度。从幅频特性和相频特性的表达式看出其与曲线位置，旋转方向不相关。貌似曲线的大小尺度对其有影响，但注意到 $\frac{l}{L}$ 可知曲线尺度也无影响。初始点选择的不同只会影响 μ，对幅频特性和相频特性同样不会产生影响。这边是傅里叶描述的强大性。

但傅里叶描述的致命缺点是必须得到对象的较好的轮廓，这在复杂背景中不是一件简单的事情。

- **基于弹性匹配图（EGM）[10,11]**

弹性图匹配在人脸识别中有很好的应用。研究实践表明:在缺失人脸二维形状信息时，从图像数据中提取多方向、多尺度(频率)的 Gabor 特征是一种合适的选择。Lades 等人首先提出了基于 Gabor 小波变换的匹配算法进行人脸识别，在此基础上，W iskott 等人又提出了弹性图匹配法(EGM)，使得计算机能自动定位关键特征点的位置。在该算法中，首先在人脸中

选取一些位置特殊的点作为特征点，比如鼻尖，眼睛，额头，下巴等;然后通过 Gabor 滤波器在特征点处滤波提取 Gabor 系数，作为特征值；将每个人脸用一组特征点及其对应的特征值来表示，并把特征值存储在称为人脸图的数据结构中；在识别过程时，先定位新的人脸图中特征点位置，然后提取特征值，生成人脸图，将其与数据库中已存的人脸图进行比较，以寻求最佳匹配。但 EGM 算法也是针对刚性的，变化少的对象其效果会比较好。

1.2.2 学习匹配算法

- **AdaBoost 加强学习算法[12,13]**

AdaBoost 加强学习算法思想是先通过简单的分类算法譬如决策树，利用少数特征对样本分类，得到弱分类器，根据分类正确的正样本率确定该弱分类器在之后构成强分类器的权重，并对错误分类的样本提高其权重构成新的训练样本传给下一个弱分类器进行训练，如此反复直到达到预定要求。然后将得到的弱分类器及相应的权重进行线性组合得到强分类器，这便称作加强型学习。

在 Viola 的论文中还提到训练多个强分类器进行级联，其中靠前的强分类器其选择特征比较粗放，靠后的特征其选择更精细。在实际的检测中真正包含人脸的子窗口一定会通过所有各层的分类器，因此在训练的时候 P 集是不需要更新的。而非人脸的子窗口会逐渐被筛选掉，并不会通过所有层，所以 N 集是需要更新的。因而用于训练后续阶段的负例集（即 N 集）是从当前检测器对非人脸图像集进行检测后判断为正例的但事实上没有目标对象（即人脸）的那些图像中获得的。

AdaBoost 算法适合于简单密集型特征，对于本身具有大量信息并对对象信息进行一定筛选的特征，其并不是最合适的。

- **SVM 支持向量机[14]**

SVM 方法的基本思想是：定义最优线性超平面，并把寻找最优线性超平面的算法归结为求解一个凸规划问题。进而基于 Mercer 核展开定理，通过非线性映射 φ，把样本空间映射到一个高维乃至于无穷维的特征空间（Hilbert 空间），使在特征空间中可以应用线性学习机的方法解决样本空间中的高度非线性分类和回归等问题。

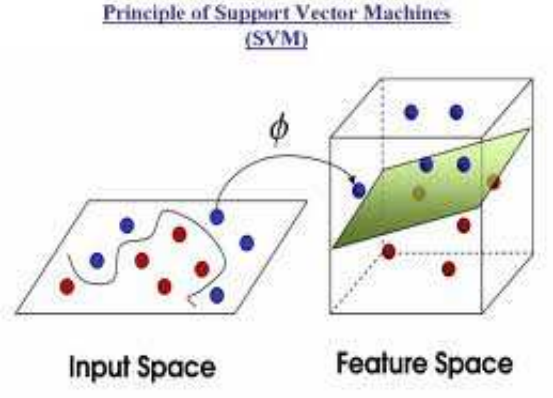


图 1.7 SVM 通过升维达到线性可分

- **NNW 人工神经网络[15]**

神经网络是一种比较暴力的学习算法，对于分类问题，其本质便是对于所给的特征，不断调整其权重，使误差达到最小。神经网络的基本单元如下：

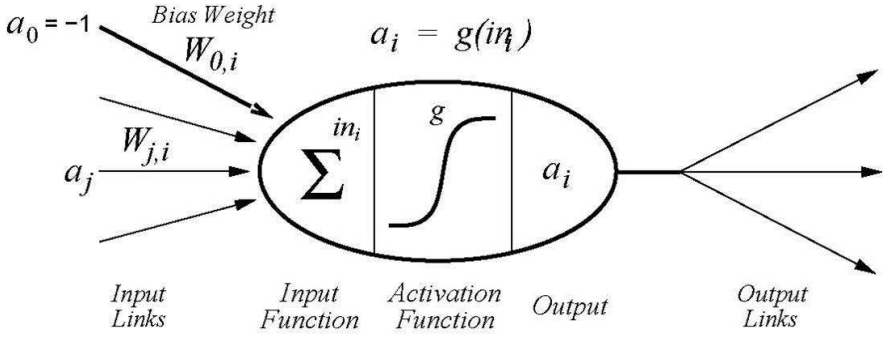


图 1.8 神经网络基本单元

输入层相当于我们的特征向量，特征向量中每个特征值具有不同的权值 W，对于激活函数一般选择阶跃函数，即相当于一个阈值，或者是 sigmoid 函数：$1/(1+e^{-x})$，输出可选择为 1 或 0，及是否为对象，然后根据结果和学习算法修改权重。常用的是梯度下降学习法。

当然这是最简单基本的神经元结构，复杂的神经网络可以包含许多个神经元，甚至包括多层结构，而不仅仅是输入输出层，可以包含隐层，而对于神经网络的具体结构是针对不同问题而异的，常用的有 BP 神经网络（Back Propagation）。

神经网络是一种比较暴力，操作简单的方法，在 Matlab 中有专门的神经网络工具包。但其收敛性很难保证，以及对隐层选择，初值选择，学习步长选择等都是基于经验公式的，没有严格的理论指导，因而在学习时带有一定的盲目性。

- **决策树分类匹配[16]**

早期决策树算法是 1986 年由 QuiIan 提出的 ID3 算法，它是一个从上到下、分而治之的归纳过程。ID3 算法选择具有最高信息增益的属性作为测试属性。

定义：设样本集 T 按离散属性 A 的 s 个不同的取值，划分为 T1，…，Ts 共 s 个子集，则 T 用 A 进行划分的信息增益为：$gain(A,T) = \inf(T) - \sum \frac{|T_i|}{|T|} \times \inf(T_i)$ 其中 $\inf(T)$ 表示 T 的信息熵。设 T 有 m 个类，则 $\inf(T) = -\sum p_j \times \log_2(p_j)$，其中 p_j 表示 T 中包含类 j 的概率。

其余的算法都是基于 ID3 算法进行优化演变而来的。譬如 CART 算法（Classification and Regression Tree）可以处理高度倾斜或多态的数值型数据，也可处理顺序或无序的类属型数据。CART 选择具有最小 gini 系数值的属性作为测试属性。

1.2.3 存在问题[17]

- **高维度的问题**

手部是由 20 个自由度以上的关节体组成的，即使手指之间存在约束而使人手的实际运动不需要 20 个自由度，研究显示参数的最少也要有 6 维。考虑到人手定位的方向性，计算量还是很大的。

- **遮盖现象**

由于手部为关节体，它的投影为大量的具有遮盖特点的形状，使得分割手部不同区域和提取有用特征的过程非常困难。目前在姿态的约束中，对于遮盖的问题，目前采取的方法是令手掌与摄像头平面平行，这样可以避免食指对其他手指的遮挡。同样也可以用于减低维数，方法是启用较低自由度的粗糙模型，对于自由度多的模型可以通过缩减自由度达到快速处理的目的。

- **处理速度问题**

面对一个图像序列，系统的也需要处理大量数据。以目前的技术设备而言，许多算法都需要大量硬件满足并行处理的需要，处理方法复杂，难以实时地识别手势。

- **无约束问题**

考虑到应用的实际要求，大部分的系统都要求适应无拘束的背景和宽范围光照的环境下。视觉输入方式的优点是对用户的运动限制较少，但由一于计算机视觉技术仍然不成熟，这种输入方式存在很多不足。把图像中的人手区域与其它区域(背景区域)划分开来始终是一个难点，特别是单目视觉情况下复杂背景下的手势分割非常困难，不仅没有成熟的理论作为指导，而且现有的方法实现困难，计算复杂度很高，效果也不是很理想。

- **快速手部运动**

手的运动一般都超过了 5m/S，手腕的转动速度也接近 360 度/s。现有的摄像机能支持 30-60HZ/帧的速率，但是现有的算法支持的帧率一般都是 20 左右，高速的手部运动和低速的采样率造成了更大的障碍。

1.3 本文研究内容

研究内容分为如下几大部分：

1、对于手的特征描述和提取。需要从现有的对人脸和物体检测算法中找到一种对手较好的特征描述，并用来训练和学习。譬如 YCrCb 肤色模型[1]，Haar_like 特征[2~4]，SIFT 特征[5~8]，FD 傅里叶描述[9]等。而本文着重于研究 HOG(Histogram of Orientation and Gradient)梯度方向直方图在手势识别及检测中的应用。

2、对提取到的 HOG 特征向量进行学习分类，达到在复杂背景下检测识别的目的。该部分内容需要对学习算法进行筛选。常用的有 AdaBoost 加强学习算法[12,13]、SVM 支持向量机[14]、ANN 人工神经网路[15]、CART 分类回归决策树[16]等。本文着重于研究 SVM 支持向量机在手势识别中的应用。

3、应用：定义不同手势对应的操作，实现在服务机器人中的应用。

1.4 本文结构

本文分为四个章节：

第一章主要介绍研究背景、意义、研究现状和存在的难题。其中在研究现状中简要的介绍了常用于研究的特征检测和学习分类方法。

第二章详细介绍了 SVM（Support Vector Machine）支持向量机学习算法的基本原理。并简单介绍了 HOG（Histogram of Orientation Gradient）梯度方向直方图算法。

第三章主要介绍如何具体实现 HOG 算法，包括平台的介绍、样本的制作、各种参数选择等；以及如何利用 LibSVM Matlab 开发包进行训练学习；最后介绍了如何结合肤色模型识别不同的手势。

第四章总结了研究成果及实验经验，探讨了存在的问题和不足及今后完善研究的方向。

二、HOG 特征与 SVM 学习算法

2.1 HOG 特征[18,19]

在介绍 HOG 特征前，需要掌握两个概念：密集型特征和稀疏型特征。稀疏特征的特点包括两点：1 包含现目标对象的更多信息；2 特征数目比较少。因而对于稀疏特征，与其相配合的是匹配算法。譬如 SIFT 特征，计算出的特征点包含了目标对象许多局部信息，如局部方向分布、主方向、尺度、亮度等，能充分放映目标对象的特征，其数量相对而言较少。相反对于密集型特征，其本身无法体现目标对象的信息，即使能体现也是隐性的，但由于其数量巨大，遍布目标对象各个位置，在这海量的信息中肯定存在能反映目标对象属性的特征。而获得这些特征的方法就需要用到学习的方法。譬如人脸识别中的 Haar_Like 特征以及稍微复杂点的 HOG 特征。

HOG 描述器是在一个网格密集的大小统一的细胞单元（dense grid of uniformly spaced cells）上计算，而且为了提高性能，还采用了重叠的局部对比度归一化（overlapping local contrast normalization）技术。

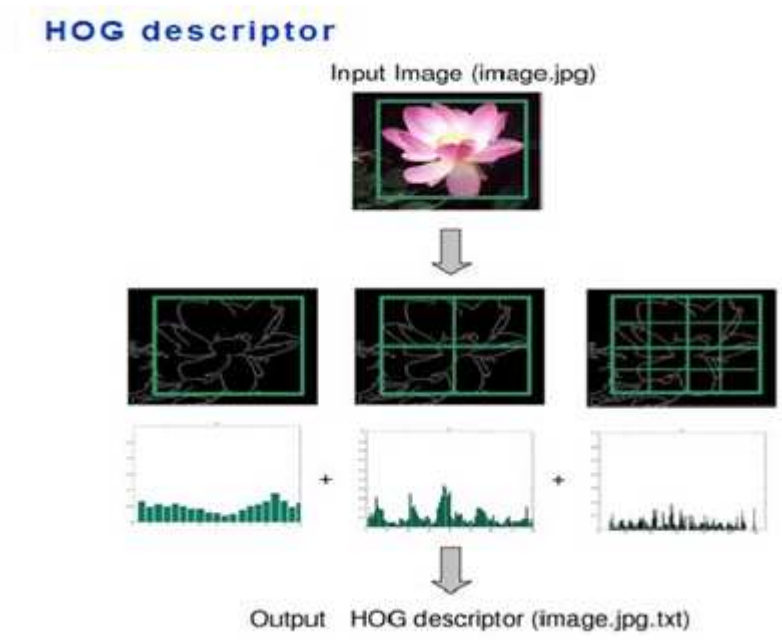


图 2.1 HOG 直方图

HOG 描述器最重要的思想是：在一副图像中，局部目标的表象和形状（appearance and shape）能够被梯度或边缘的方向分布很好地描述。具体的实现方法是：首先将图像分成小的连通区域，我们把它叫细胞单元。然后采集细胞单元中各像素点的梯度的或边缘的方向直方图。最后把这些直方图组合起来就可以构成特征描述器。为了提高性能，我们还可以把这些局部直方图在图像的更大的范围内（我们把它叫 block）进行对比度归一化。通过这个归一化后，能对光照变化和阴影获得更好的效果。具体可以参考图 2. 1。

算法的实现：

- 图像预处理：包括是否 Gamma 校正，是否平滑滤波。
- 梯度的计算（Gradient computation）

最常用的方法是：简单地使用一个一维的离散微分模板（1-D centered point discrete derivative mask）在一个方向上或者同时在水平和垂直两个方向上对图像进行处理。作者也尝试了其他一些更复杂的模板，如 3×3 Sobel 模板，或对角线模板（diagonal masks），但是在这个行人检测的实验中，这些复杂模板的表现都较差，所以作者的结论是：模板越简单，效果反而越好。作者也尝试了在使用微分模板前加入一个高斯平滑滤波，但是这个高斯平滑滤波的加入使得检测效果更差，原因是：许多有用的图像信息是来自变化剧烈的边缘，而在计算梯度之前加入高斯滤波会把这 些边缘滤除掉。

- 构建方向的直方图（creating the orientation histograms）

第三步就是为图像的每个细胞单元构建梯度方向直方图。细胞单元中的每一个像素点都为某个基于方向的直方图通道（orientation-based histogram channel）投票。投票是采取加权投票（weighted voting）的方式，即每一票都是带权值的，这个权值是根据该像素点的梯度幅度计算出来。可以采用幅值本身或者它的函数来表示这个权值，实际测试表明： 使用幅值来表示权值能获得最佳的效果。

- 归一化

把细胞单元组合成大的区间（grouping the cells together into larger blocks），由于局部光照的变化 （variations of illumination）以及前景-背景对比度（foreground-background contrast）的变化，使得梯度强度（gradient strengths）的变化范围非常大。这就需要对梯度强度做归一化，作者采取的办法是：把各个细胞单元组合成大的、空间上连通的区间（blocks）。 这样以来，HOG 描述器就变成了由各区间所有细胞单元的直方图成分所组成的一个向量。这些区间是互有重叠的，这就意味着：每一个细胞单元的输出都多次作用于最终的描述器。作者还发现，在对直方图做处理之前，给每个区间（block）加一个高斯空域窗口（Gaussian spatial

window）是非常必要的，因为这样可以降低边缘的周围像素点（pixels around the edge）的权重。作者采用 L2-Hys 归一化方式，它可以通过先进行 L2-norm，对结果进行截短（clipping），然后再重新归一化得到。

- SVM 分类器（SVM classifier）

最后一步就是把提取的 HOG 特征输入到 SVM 分类器中，寻找一个最优超平面作为决策函数。可以将每个 Cell 的直方图中 9 个数据排成一个列向量，作为 SVM 学习的特征向量。其维度常常上万，但是由于 SVM 学习算法关心的不是特征向量的维度，而仅仅是样本的数量，因而这并不影响 SVM 的正常学习。

2.2 SVM 学习[14,20]

2.2.1 线性分类器

在介绍 SVM 支持向量学习之前需要先了解线性分类器，SVM 实际是感知机扩展。

我们以二值分类器作为讨论对象，对于训练样本：

$$\begin{aligned} &X_i = (x_{i1}, x_{i2}...x_{in}) \\ &Y_i = y_i, y_i \in \{-1,1\} \\ &i = 1...l \end{aligned} \tag{2.1}$$

其中 X_i 称为分类对象的特征向量，如计算的 HOG 特征向量；Y_i 为理想输出，表征所属类别。现在我们需要构造一个分类器 $g(X_i) = \text{sgn}(f(X_i))$ 使 $X_i \xrightarrow{g} Y_i$，若 $f(X_i)$ 有以下形式：

$$f(X_i) =< w, X_i > +b \tag{2.2}$$

则称为线性分类器，其中 w 为权值向量，b 为偏置。

若输入为二维，线性分类器相当于一条直线把平面上的点分为两部分；若输入为三维，线性分类器相当于平面把空间上的点分为两部分；推广到 n 维，则相当于一个超平面把 n 维中的点分为两部分。

然而线性分类器有很大的局限性，尤其在实际应用中直接线性可分的情况非常少，SVM 在引入核函数后大大增强了线性分类器的能力，这将在后文看到。

2.2.2 结构风险及泛化误差界

对于一个分类器的性能评价分为两部分：

- 经验风险：表征分类器在给定训练样本上的误差
- 结构风险：表征在多大程度上可以信任分类器在未知对象上分类的结果，是分类器泛化能力的体现

这二者是相互制约的，当经验风险很小时，可能导致过学习状态，待分类正样本与训练正样本微小差异都可能导致分类失败，其泛化能力会很差。相反结构风险很小时，可能导致分类器将待分类负样本误认为是正样本。

而泛化误差界表征真实风险，为经验风险与结构风险之和。为了得到性能良好的分类器并须在经验风险和结构风险之间进行权衡，使真实风险达到最小。而在 SVM 中，通常能做到经验风险接近 0，其目的在于最小化结构风险。

2.2.3 几何间隔

几何间隔的引入正是为了解决最小化结构风险。其定义如下：

$$\delta_i = y_i(<\frac{w}{\|w\|_2}, X_i> + \frac{b}{\|w\|_2}) \tag{2.3}$$

由于$|y_i|=1$，若取几何间隔绝对值可以发现，这是点到直线距离公式的推广。另外，若几何间隔大于零表示正确分类，否则表示分类错误。

这里引入 Novikoff 提出的定理：训练样本 S 满足 y_i 不全相等，且存在$\|w_e\|_2=1$，满足 $Y(<w_e, X>+b_e) = Y(<\frac{w}{\|w\|_2}, X>+\frac{b}{\|w\|_2}) \geq \delta = \frac{r}{\|w\|_2}$，其中 $r=\max|r_i|, r_i = y_i(<w, X_i>+b)$，令 $R=\max\|X\|_2$，则 S 上最大误分次数：$(\frac{2R}{\delta})^2 = (\frac{2\|w\|_2 R}{r})^2$。

于是最小化结构风险相当于最小化最大误分次数，即等价于最大化 δ。一般我们规定 r=1，于是问题等价于最小化 $\|w\|_2$。于是最小化结构风险的分类问题等价于：

$$\begin{aligned} &\min && f = \frac{1}{2}\|w\|_2^2 \\ &subject\ \ to && g_i = y_i(<w, X_i>+b) - 1 \geq 0 \end{aligned} \tag{2.4}$$

2.2.4 核函数

2.2.3 节最终将线性分类问题归结为一个优化问题，但该优化问题是否有最优解无法确定。很显然对于线性不可分的问题上述优化问题不存在最优解，其优化毫无意义。这也正是线性分类器所遇到的瓶颈。为了解决分类器的性能，引入了核函数。

为了了解核函数，我们先看一个分类实例：

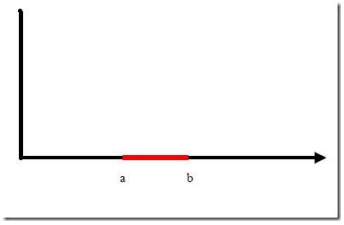


图 2.2 线性不可分实例

我们把横轴上端点 a 和 b 之间红色部分里的所有点定为正类，两边的黑色部分里的点定为负类。试问能找到一个线性函数把两类正确分开么？不能，因为二维空间里的线性函数就是指直线，显然找不到符合条件的直线。

但我们可以找到一条曲线，例如下面这一条：

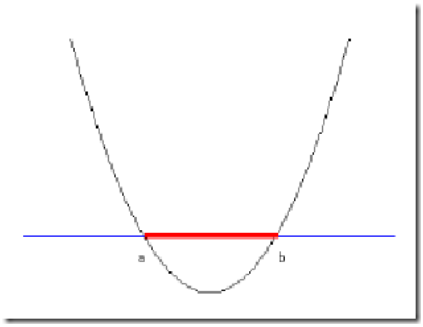


图 2.3 高维后线性可分实例

显然通过点在这条曲线的上方还是下方就可以判断点所属的类别。其表达式可以写为：

$$g(x)=c_0+c_1x+c_2x^2 \tag{2.5}$$

问题只是它不是一个线性函数，但是，新建一个向量 y 和 a：

$$y=\begin{bmatrix}y_1\\y_2\\y_3\end{bmatrix}=\begin{bmatrix}1\\x\\x^2\end{bmatrix},a=\begin{bmatrix}a_1\\a_2\\a_3\end{bmatrix}=\begin{bmatrix}c_0\\c_1\\c_2\end{bmatrix} \tag{2.6}$$

这样 $g(x)=f(y)=ay$,在任意维度的空间中，虽然其中的 a 和 y 都是多维向量，但因为自变量 y 的次数不大于 1，这种形式的函数依旧是一个线性函数。

原来在二维空间中一个线性不可分的问题，映射到四维空间后，变成了线性可分的！因此这也形成了我们最初想解决线性不可分问题的基本思路——向高维空间转化，使其变得线性可分。但是要找到这种映射并没有系统的理论与方法来寻找和构造。

但是这并不表示穷途末路，为了解决这一难题须引入权重的对偶形式。定义权重对偶形式：$w=\sum_{i}^{l}\alpha_i y_i X_i$，其中 L 表示样本数目。这种定义是合理的，影响权重的因素就是输入样本及其对应的输出。于是分类器表达式为：

$$g(X_j)=\mathrm{sgn}(<w,X_j>+b)=\mathrm{sgn}(<\sum_{i}^{l}\alpha_i y_i X_i,X_j>+b)=\mathrm{sgn}(\sum_{i}^{l}\alpha_i y_i <X_i,X_j>+b) \quad (2.7)$$

假设存在映射到高位的函数 $\Phi(X)$，于是分类器表达式变为：

$$g(X_j)=\mathrm{sgn}(\sum_{i}^{l}\alpha_i y_i <X_i,X_j>+b)=\mathrm{sgn}(\sum_{i}^{l}\alpha_i y_i <\Phi(X_i),\Phi(X_j)>+b) \quad (2.8)$$

由上式可知我们真正关注的不是低维的向量映射到高维空间后向量的具体形式，我们更关心两个低维向量映射到高维空间后对应向量的内积 $<\Phi(X_i),\Phi(X_j)>$。于是我们有了核函数的定义：核函数是一个函数 K，对所有 $X,Z\in\Omega$，有 $K(X,Z)=<\Phi(X),\Phi(Z)>$，其中 $\Phi(X)$ 是将低维空间的向量映射到高维空间的函数。

于是我们通过将低维线性不可分的样本映射到高维后，转变为线性可分的问题，其数学表达式见式 2.。而在实际操作中我们并不需要知道这个映射的具体表达式，而只需要知道映射到高维后向量的内积，便可计算 $<w,\Phi(X_i)>$，而这需要依靠核函数。

$$\begin{aligned} &\min && f=\frac{1}{2}\|w\|_2^2 \\ &subject\quad to && g_i=y_i(<w,\Phi(X_i)>+b)-1\geq 0 \end{aligned} \quad (2.9)$$

核函数的构造与证明比较复杂，需要用到 Mercer 定理，这里不做细述，仅列举常用的核函数模型：

线性核函数：$$K(X_i,X_j)=<X_i,X_j> \quad (2.10)$$

多项式核函数：$$K(X_i,X_j)=(<X_i,X_j>+1)^d \quad (2.11)$$

径向基核函数：$$K(X_i, X_j) = e^{(-\frac{\|X_i - X_j\|_2^2}{\sigma^2})} \tag{2.12}$$

S 型核函数：$$K(X_i, X_j) = \tanh(\beta < X_i, X_j > + r) \tag{2.13}$$

本论文中用来学习的是线性核函数。

2.2.5 凸优化及拉格朗日法

引入凸优化的目的是为了在理论上证明上述 2.2.4 节中提出的优化是肯定存在最优解的。先了解一下几个定义：

二次规划：目标函数 f 是二次的，而约束 g 均为线性约束（具有 g=ax 形式）的优化问题。

仿射函数：可用某个举证 A 和响亮表示为 $f(x) = Ax + b$

凸集合 Ω：$\forall w, u \in \Omega, \theta \in (0,1) \Rightarrow (\theta w + (1-\theta)u) \in \Omega$

凸函数 f：$\forall w, u \in \Omega, \theta \in (0,1) \Rightarrow f(\theta w + (1-\theta)u) \le \theta f(w) + (1-\theta) f(u)$

凸函数具有良好的性质：其局部最小值也是全局最小值。

凸优化：优化问题的集合为凸集合，目标函数和所有约束均为凸函数的优化问题。

凸优化有个重要的性质，便是凸优化必有最优解。

而 SVM 优化问题中集合为凸集合，目标函数为凸二次函数，所有约束为凸一次函数，故 SVM 优化问题属于凸优化问题，其必有最优解。

对于求式 2.最优解，可以利用广义拉格朗日法求解。

构造拉格朗日式：

$$L(w, b, \beta) = \frac{1}{2} < w, w > - \sum_j^l \beta_j [y_j (< w, \Phi(X_j) > + b) - 1] \tag{2.14}$$

其中 β 为拉格朗日因子。

分别对 w 与 b 求偏导等于零有：

$$\frac{\partial L(w, b, \beta)}{\partial w} = w - \sum_j^l \beta_j y_j \Phi(X_j) = 0 \Rightarrow w = \sum_j^l \beta_j y_j \Phi(X_j) \tag{2.15}$$

$$\frac{\partial L(w, b, \beta)}{\partial b} = \sum_j^l \beta_j y_j = 0 \tag{2.16}$$

将式 2.带回式 2.，可得到：

$$L(w,b,\beta)=\sum_{j}^{l}\beta_j-\frac{1}{2}\sum_{j}^{l}\beta_j y_j\beta_i y_i<\Phi(X_j),\Phi(X_i)> \tag{2.17}$$

令：

$$Q_{ij}=y_j y_i<\Phi(X_j),\Phi(X_i)> \tag{2.18}$$

于是得到以下等价优化形式：

$$\begin{aligned}&\min && \frac{1}{2}\beta^T Q\beta-e^T\beta\\ &subject && to \quad y^T\beta=0\end{aligned} \tag{2.19}$$

对于上式的解法具体可参见林智仁的 *a library for support vector machines*，这儿不在细述。

2.2.6 松弛变量

最后再介绍下松弛变量。虽然我们通过映射将低维的线性不可分问题变为高维的线性可分问题，但这所说的线性可分是针对样本总体来说的。对于映射到高维后，仍然可能存在正样本存在于理应是负样本的区域中。这是无可避免的，尤其在考虑成千上万的数据时，数据样本本身就存在偏差，存在随机性，存在特殊性。对于这种情况我们不能因为一两各个例就终止我们的学习分类，这时我们可以引入松弛变量，对于分类不要求每个样本都分类正确，允许存在误差。体现在数学表达式中为：

$$y_i(<w,\Phi(X_i)>+b)\geq 1-\xi_i \tag{2.20}$$

引入松弛变量后的优化数学模型为：

$$\begin{aligned}&\min && \frac{1}{2}\|w\|_2^2+C\sum_i\xi_i\\ &subject && to \quad y_i(<w,\Phi(X_i)>+b)\geq 1-\xi_i\end{aligned} \tag{2.21}$$

其解法同为加入松弛变量时的模型。其中 C 称为惩罚因子，表征你有多么重视离群点，C 越大越重视，越不想丢掉它们。

三、具体实现及效果

方案的实现的基本思路包含两个部分：检测+检测前提下的识别。为了简化对于手检测的难度，我们规定检测器只检测手的初始状态（Open 状态）即手张开的状态。对于初始状态的手的检测是一个传统分类中的二值分类问题。

检测到张开状态（Open 状态）下的手后再识别不同手势，比直接识别手势简单许多。举个具体的例子，譬如我们可以得到不同手势对于 A 特征具有不同特征值，但我们无法仅仅根据 A 特征的不同值来推断当前对象是不同的手势，因为具有相同特征值的对象可以完全不是手。预期实现的手势识别见图 3.1。

方案的总流程：

- 初始化，定义检测频率，检测窗口等参数。
- 检测 Open 状态的手，确定手的矩形区域 Rect
- 由定义的检测频率，在检测的时间间隔内由肤色模型和画圆法识别不同手势。
- 当时间等于规定的检测间隔时间，返回第二步重新检测 Open 状态的手，更新手的矩形区域 Rect。

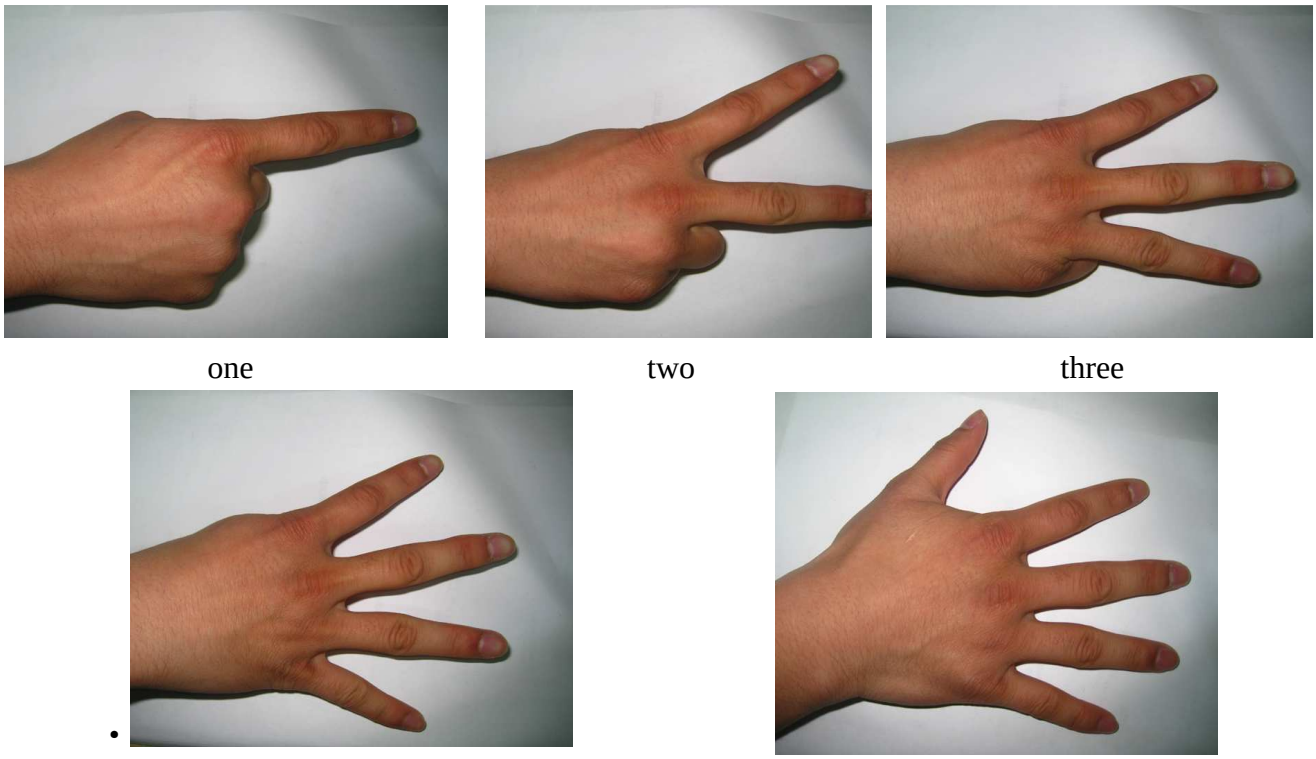


图 3.1 预期识别的手势

3.1 开发平台

OpenCV2.0：微软开源计算机视觉库，OpenCV 高效的矩阵运算能力以及对通用视觉算法的完美实现，都有助于我们项目的开发（了解更多关于 OpenCV，可登陆 OpenCV 中文论坛 http://www.opencv.org.cn）。

LibSVM：是台湾大学林智仁(Lin Chih-Jen)副教授等开发设计的一个简单、易于使用和快速有效的 SVM 模式识别与回归的软件包。本文利用 LibSVM 的 MATLAB 版本，便于获得学习模型的相关参数（了解更多 LibSVM MATLAB 版本，可登陆 MATLAB 中文论坛 http://www.ilovematlab.cn/index.php）。

Directshow：流媒体处理开发包，进行高效实时的视频采集。另一方面由于 OpenCV 的 HighGUI 库只提供了 VFW 接口的摄像头支持，而许多摄像头却是 Directshow 驱动的，为了解决 OpenCV 对 Directshow 驱动的摄像头的兼容性问题，所以选择了 Directshow 开发包。

Visual Studio 2008：作为代码编写与编译的平台，其开发语言为 C++。

Matlab 2010：作为 LibSVM 执行及相关学习代码编写与编译的开发平台，选择 LibSVM MATLAB 版，还考虑到 MATLAB 对于数据友好的图形表示。

Photoshop CS5：Photoshop 主要用于手势数据库的制作。

3.2 数据库制作

对于基于学习方法的识别检测问题，拥有大量可学习的样本是至关重要的。这里提供几个可用的人体数据库：

CVonline Databases: http://homepages.inf.ed.ac.uk/cgi/rbf/CVONLINE/entries.pl?TAG363

Korean Intelligent Media Lab Database: http://imlab.postech.ac.kr/

University of Dublin Face Database: http://dsp.ucd.ie/~prag/

Video sequences of American Sign Language (ASL): http://csr.bu.edu/asl/html/sequences.html

MIT Computer Science and Artificial Intelligence Lab Database: http://db.csail.mit.edu/

虽然有不少人体数据库，但大部分是关于人脸及行人的，只有少量手势数据库，其通用性也不强，因而需要制作属于自己的数据库。下面具体介绍下如何制作自己的手势数据库，及制作过程中需要注意的事项。

3.2.1 原始图像采集

可利用普通的数码相机在不同环境和背景下采集尽量多样的样本（见图 3.2），本文控制在 70~100 张。本文以张开的手作为识别对象，采集样本时须注意：

- 保证手势的统一，其空间变化性不要过大，限制左右旋转角度、手的开合程度，另外不要有遮挡。对于原始样本的大小不做要求，之后会做标准化处理。
- 在保证手势的尽量一致性后，要注意背景、光照的多样性，尽可能多的变换场景，使基于该样本集学习后的模型具有一定的鲁棒性。背景可尝试用不同的彩色图片（如杂志封面），简单而实用。
- 建议在采集正样本时，没必要采集不同人的手。由于不同的人其手的差异性也挺大的，在增强模型的鲁棒性同时会增加学习的难度和学习样本标准化的难度。仅作为建议，是因为这仅是实际经验之谈，没有具体的理论支持。

(a)

(b)

(c)

(d)

图 3.2 (a)为晚上，(b)为白天室外，(c)为白天室内，(d)为用彩色图片作为背景

3.2.2 原始样本中分割出手

本文利用 OpenCV 的库函数编写了两个程序：ObjectMaker 和 GetPositive，其作用是达到批量处理图像的目的。

ObjectMaker 可以记录鼠标所框选的矩形区域，并保存与 info.txt 文本中，info 的格式如下：rawdata/034.bmp 1 211 145 319 314，分别表示样本地址，样本中有手的数量，样本中记录手的矩形框的左上角坐标和矩形框宽度和高度。

GetPositive 可以读取 info.txt 中的信息，利用 cvSetImageROI 将 info 中记录的关于手的感兴趣区域分割保存，并可以标准化所有分割出的手的大小。本论文限定分割出的手大小为 128*128 像素。

分割后得到的样本如下：

(a) (b) (c) (d)

图 3.3 分割后的样本

3.2.3 制作正样本

利用 Photoshop 中的磁套功能可以得到样本中手的轮廓，利用磁套功能时精确度要求不高。获得手的轮廓的目的是为了用软件产生更多的正样本，而免去了到处拍摄正样本。

利用 Photoshop 中记录动作和批处理功能，本文将 1000 张非手的图片统一为 144*144 像素的.jpg 格式图片，其中 400 张作为训练的负样本，100 张作为测试的负样本，500 张作为产生正样本的背景图片（400 张用来训练，100 张用来测试）。

将每个原始样本中得到的手的轮廓与背景图片中的 10 张组合，制作出更多的正样本，利用这种方法完全不需要担心样本数量的不够。

需要提到的是在利用磁套获得轮廓时注意保存为 photoshop 图片格式 psd，否则无法保留图片的图层信息，将会使轮廓与白色的背景合为一体，因而无法达到轮廓与不同背景图片融合的目的。这样做出的正样本可以保证手都在样本的正中间。

3.2.4 样本分类

TrainPositive：用于训练制作的 400 张正样本

TrainNegative：用于训练制作的 400 张负样本

TestPositive：用于模型测试制作的 100 张正样本

TestNegative：用于模型测试制作的 100 张负样本

DetectSample：用于检测的 300 张样本，其中 100 张为拍摄的较友好的正样本，100 张为拍摄的较恶劣的正样本，100 张负样本。所谓的较友好的正样本，指的是手势的变化不大；较恶劣的正样本，指的是变化较大的样本，其变化包括手的开合程度，平面左右旋转，空间左右侧旋，空间前后仰俯，光线变化等。检测样本中的图片大小没有限制，后文将提到在不同尺度上检测手。

(a) (b) (c)

图 3.4 (a)手的轮廓，(b)背景图片或负样本，(c)制作的正样本

(a) (b) © (d)

图 3.5 (a)友好的检测正样本，(b)(c)恶劣的检测正样本，(d)检测负样本

3.3HOG 特征计算与提取

计算 HOG 及检测目标是否存在的参数如下：

表 3.1 HOG 特征计算参数列表

InputImage	3 通道 RGB 图像，144*144 用于训练和测试，>144*144 用于检测
GammaCorrection	False，本实验中不需要 Gamma 校正降低图像明暗对比度
SmoothingFilter	False，本实验中不需要平滑滤波，相反要最大限度保留边缘信息
WindowSize	144*144，检测窗口大小即为训练样本大小
BlockSize	16*16，每个检测窗口包含 289 个 Block
BlockStride	8，Block 以 8 个像素为单位，水平或者垂直移动
CellSize	8*8，每个 Block 包含 4 个 Cell
Histogram Bins	9, 即将 0~180° 分为 9 个 bin，以 Cell 为单位进行统计
Gaussion Spatial Window	8，二维高斯函数中标准差，为 Block 大小的 1/2，以 Block 为单位用于分配 Block 中不同像素的权重
Normalization	L2Hys，以 Block 为单位进行归一化，具体见后文
Gradient Filter	[-1,0,1]，计算像素梯度算子，左右相邻像素之差
WindowStride	8，DetectWindow 以 8 个像素为单位，水平或者垂直移动
PaddingSize	0，本文中不对图像进行边缘扩充

3.3.1GammaCorrection 和 Smoothing

现实世界中几乎所有的 CRT 显示设备、摄影胶片和许多电子照相机的光电转换特性都是非线性的，计算机绘图领域惯以此屏幕输出电压与对应亮度的转换关系曲线，称为伽玛曲线。所谓伽玛校正就是对图像的伽玛曲线进行非线性化补偿，实现显示图像与实际图像区域对比度相同的目的。本文中对于对象的明暗对比度要求不高，可以选择不进行 Gamma 校正。考虑到 Gamma 曲线是指数形式的，若进行 Gamma 校正本文将采用开方的方法。

图像平滑是指用于突出图像的宽大区域、低频成分、主干部分或抑制图像噪声和干扰高频成分，使图像亮度平缓渐变，减小突变梯度，改善图像质量的图像处理方法。图像平滑会丧失关于边缘的信息，使边缘模糊化。对于手的检测识别而言，边缘轮廓信息是至关重要的，HOG 本身就是基于梯度的变化，因而必须禁止平滑滤波。

3.3.2 梯度计算

梯度方向直方图，顾名思义，检测效果必然对于梯度的计算非常敏感。梯度计算方式：

$$\begin{aligned} Gradient &= \sqrt{dx^2 + dy^2} \\ Orientation &= \arctan(\frac{dx}{dy}) \end{aligned} \tag{3.1}$$

关键在于其中 dx 与 dy 如何选择，常用的算子包括：

- 一维算子：[-1,1]或[-1,0,1]或[-1,-8,0,8,1]。其中 dx 与 dy 是相同的算子，只是一个水平，一个垂直计算。
- 二维算子：$dx = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}, dy = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$ 或者 $dx = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}, dy = \begin{pmatrix} -1 & -2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & 1 \end{pmatrix}$，二维算子对应的矩阵或者说 Mask 甚至可以更大

考虑到 HOG 算法的侧重点是局部信息，因而对于 Mask 很大的二维算子反而不适合，本文中用最简单的一位算子[-1,0,1]

具体计算梯度时还要考虑一下两个问题：

- 多通道：对于 RGB 的 3 通道彩色图像，其梯度计算是 3 个通道独立进行，再在三者中取最大值作为该点的梯度值
- 高斯权重：直方图统计时，根据每个像素的 Orientation 大小确定其所属的 bin，而 bin 中记录表示该点方向程度的量即为 Gradient，可以说 Gradient 的大小就是决定着 Cell 中的主方向。而高斯权重是说在每个 Cell 中，并不是所有像素对 Cell 主方向或者直方图的贡献的权重都是等值的。高斯权重规定 Cell 中靠近 Block 中心的像素对 Cell 直方图的贡献更大，而 Cell 中远离 Block 中心的像素点对于 Cell 直方图的贡献更廉价。高斯权重表现了对像素位置的偏向性，我们更关注那些聚集于 Block 中心的像素，因为他们更能表现 Block 的局部信息与特征，所以在统计 Cell 中的直方图时其权重也更大。我们选择二维高斯分布作为梯度的权重分布：

$$G(dx, dy) = e^{-\frac{dx^2 + dy^2}{2\sigma^2}} \tag{3.2}$$

式中 dx 与 dy 表示在以 Block 为单位的局部坐标中，以 Block 中心(8,8)为原点，各像素与原点的距离，注意与梯度计算时的 dx 区分，这里不是亮度值之差。

3.3.3 直方图统计

这是 HOG 算法最关键的一步，在计算出每个像素的 Orientation，我们需要对 Orientation 运用所谓的三插值算法（Trilinear Interpolation）。首先我们先介绍一维线性插值算法：规定 $x_i(i=1\sim 9)$ 表示直方图中 9 个 bin 的中心（如 x_1=10），$h(x_i)$ 表示第 i 个 bin 中统计的值（如前文所说可以是高斯权重函数与梯度值之乘积 w），bin 的宽度为 b。现在我们计算某个像素的 Orientation 为 x，且 $x_i < x < x_{i+1}$，当我们要更新 $h(x_i)$ 时，可以采用如下公式：

$$h(x_i) \leftarrow h(x_i) + w(1-\frac{x-x_i}{b}) \tag{3.3}$$

$$h(x_i) \leftarrow h(x_i) + w(\frac{x-x_i}{b}) \tag{3.4}$$

很显然我们可以推广到 3D 的情况，其公式如下：

$$\begin{cases} h(x_1,y_1,z_1) \leftarrow h(x_1,y_1,z_1) + w(1-\frac{x-x_i}{b_x})(1-\frac{y-y_i}{b_y})(1-\frac{z-z_i}{b_z}) \\ h(x_2,y_1,z_1) \leftarrow h(x_2,y_1,z_1) + w(\frac{x-x_i}{b_x})(1-\frac{y-y_i}{b_y})(1-\frac{z-z_i}{b_z}) \\ h(x_1,y_2,z_1) \leftarrow h(x_1,y_2,z_1) + w(1-\frac{x-x_i}{b_x})(\frac{y-y_i}{b_y})(1-\frac{z-z_i}{b_z}) \\ h(x_1,y_1,z_2) \leftarrow h(x_1,y_1,z_2) + w(1-\frac{x-x_i}{b_x})(1-\frac{y-y_i}{b_y})(\frac{z-z_i}{b_z}) \\ h(x_2,y_2,z_1) \leftarrow h(x_2,y_2,z_1) + w(\frac{x-x_i}{b_x})(\frac{y-y_i}{b_y})(1-\frac{z-z_i}{b_z}) \\ h(x_2,y_1,z_2) \leftarrow h(x_2,y_1,z_2) + w(\frac{x-x_i}{b_x})(1-\frac{y-y_i}{b_y})(\frac{z-z_i}{b_z}) \\ h(x_1,y_2,z_2) \leftarrow h(x_1,y_2,z_2) + w(1-\frac{x-x_i}{b_x})(\frac{y-y_i}{b_y})(\frac{z-z_i}{b_z}) \\ h(x_2,y_2,z_2) \leftarrow h(x_2,y_2,z_2) + w(\frac{x-x_i}{b_x})(\frac{y-y_i}{b_y})(\frac{z-z_i}{b_z}) \end{cases} \tag{3.5}$$

虽然公式如上，但实际运用时却有所不同。首先我们需要明确对于每个像素运用 Trilinear Interpolation 时其对应的 3 个维度为，在具体操作时每个维度其实可以单独考虑的。

- 同一个 Cell 中直方图相邻两个 bin，如一维线性插值法所举实例
- 水平方向相邻 Cell 中不同 Histogram 中同一个 bin

- 垂直方向相邻 Cell 中不同 Histogram 中同一个 bin

其中第一个维度，在计算每点梯度和方向时将会同时进行插值。关于其中第二维、三维较为为晦涩。编程实现的思路具体参考下图说明及分析：

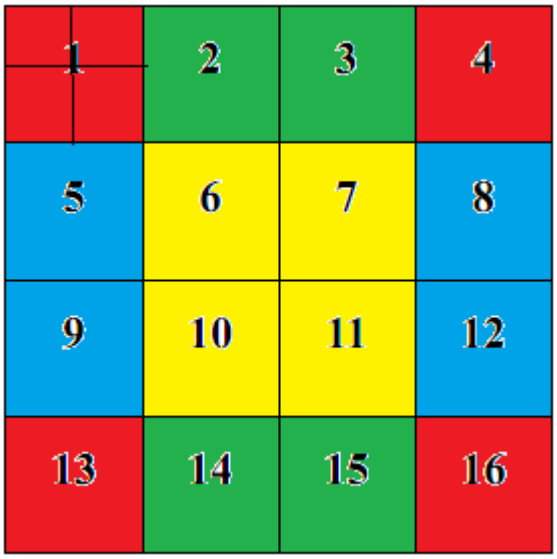


图 3.6 Trilinear 插值后两维插值示意图

容易看出 1,2,3,4 单元属于 Cell1，3,4,7,8 单元属于 Cell2，9,10,13,14 单元属于 Cell3，11,12,15,16 单元属于 Cell4。具体编程时，将 Block（16*16）分为四个区域，如图 3.6 四种不同颜色所示，每个区域包含 4 个 4*4 像素大小的矩形单元，并规定 Block 中中心像素（8,8）为原点，其余像素取相对坐标（x,y）。对各区域进行直方图统计时：

- 红色区域：认为该区域的像素点只对其所属 Cell 的直方图统计有贡献，不进行插值。
- 绿色区域：认为该区域的像素点对水平相邻的两个 Cell 的直方图统计均有贡献，即进行水平方向维度插值。如 2 单元中像素（x,y）对 Cell1 和 Cell2 中直方图统计都有贡献，但对 Cell3 和 Cell4 无影响，对 Cell1 与 Cell2 权重可由一维线性插值公式得到分别为：

$$(1-\frac{x-(-8)}{16}) \text{ 和 } (\frac{x-(-8)}{16})$$

- 蓝色区域：认为该区域的像素点对垂直相邻的两个 Cell 的直方图统计均有贡献，即进行垂直方向维度插值，其分析同绿色区域的分析。
- 黄色区域：认为该区域的像素点对水平和垂直相邻的 Cell 的直方图统计均有贡献，即进行水平和垂直维度插值。如 6 单元中像素（x,y）对 Cell1、Cell2、Cell3、Cell4 的直方图统计均有贡献，其权重可有二维线性插值公式得到分别为：

$$(1-\frac{x-(-8)}{16})(1-\frac{8-y}{16})、(\frac{x-(-8)}{16})(1-\frac{8-y}{16})、(1-\frac{x-(-8)}{16})(\frac{8-y}{16}) \text{ 和 } (\frac{x-(-8)}{16})(\frac{8-y}{16})$$

3.3.4 归一化

在计算完每个像素的梯度和方向，且完成直方图统计后，需要以 Block 为单位进行归一化。归一化能增强局部信息对光照、背景的鲁棒性。将每个 Cell 的直方图中 9 个 bin 的值构成一个 36 维的向量 v，这便是我们归一化的对象。常用的归一化有：

- L2-Norm: $v \to v / \sqrt{\|v\|_2 + \varepsilon^2}$ (3.6)
- L2-Hys: $v \to \max(v / \sqrt{\|v\|_2 + \varepsilon^2}, \alpha)$ (3.7)
- L1-Norm: $v \to v / (\|v\|_1 + \varepsilon)$ (3.8)
- L1-Sqrt: $v \to v / \sqrt{\|v\|_1 + \varepsilon^2}$ (3.9)

式中：

$\|v\|_1$——表示向量的一阶范式，$\|v\|_1 = \sum |v_i|$

$\|v\|_2$——表示向量的二阶范式，$\|v\|_2 = \sqrt{\sum v_i^2}$

ε ——为一固定的可调参数，可用来限制向量最大值，在本文中取 0.1*16*16=25.6

α ——为一固定的可调参数，用来限幅，在本文中取 0.2

本文中采用 L2-Hys 归一化，并将归一化的 v，再进行一次单位化即：$v \to v / \sqrt{\|v\|_2}$

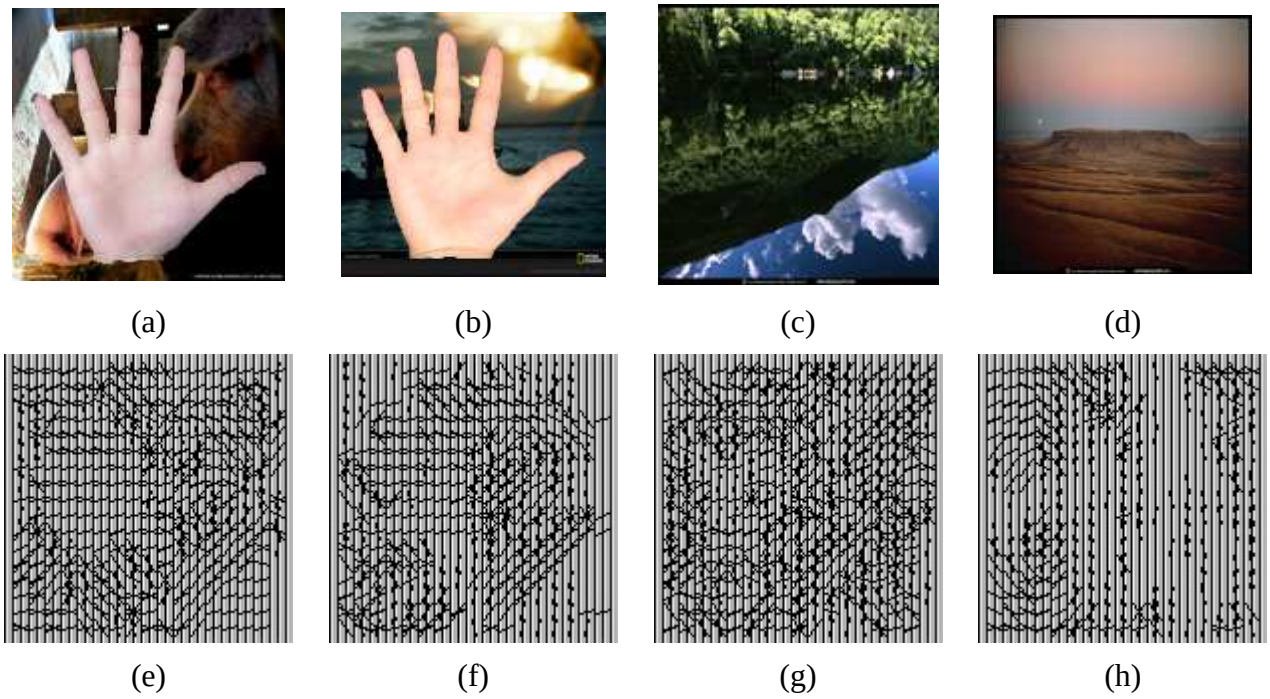
3.3.5 HOG 特征向量计算流程

表 3.2 HOG 算法流程

Input：144*144，RGB，3 通道图像 img Compute Gradient：计算 img 中每个像素的 Gradient 和 Orientation（其中 Gradien 和 Orienation 保存为 144*144,2 通道 cv::Mat 矩阵形式），并进行相邻 bin 间的插值分别保存在 2 个通道中。 Initialization：初始化每个 Block 相对 img 原点偏置，计算 Block 中每个像素二维插值对于不同 Cell 直方图的权重 For 循环：由记录的偏置信息提取 Block 的位置 For 循环：确定 Block 位置后，计算每个 Block 中 36 个 bin 的信息 Output:289*36=10404 维的特征向量，记录在.txt 文件中

3.3.6 HOG 特征形象显示

在程序编写时，额外编写了部分代码，以 Cell 为单位，每个 Cell 中画一条直线，直线倾斜的角度正是直方图最大值所对应的角度，形象的表示出 HOG 特征，见图：



(a) (b) (c) (d)

(e) (f) (g) (h)

图 3.7 a~d 分别为正负样本，e~h 为其对应 HOG 图

从 HOG 图中多少可以感性的认识到正样本还是具有一定共性的，而负样本则相差比较大。具体具有什么样的共性特征则需要 SVM 学习机学习。

3.4 LibSVM 学习分类

利用 Matlab 中 textread 函数读取.txt 中数据（这里要提一点，数据存储时每个值用空格隔开，每幅图片对应的特征向量用换行符隔开，这样利用 Matlab 读取数据时，每行数据对应一个行向量）。然后构造训练数据 TrianData 和理想输出 TrainLabel。最后调用 LibSVM 中的 svmTrain，其中‘-S 0’表示选择 C-SVM 二值分类，‘-t 0’表示选择 Linear 核函数，‘-C 1’表示惩罚因子大小，惩罚因子越大表示越不能接受错误分类。具体代码见表 3.3

model 相关参数可见图 3.8，其中 rho 相当于决策函数中 $h(x)=\mathrm{sgn}(wx+b)$ 的-b；sv_coef 相当于权重 $w=\sum_i \alpha_i y_i x_i$ 中的 $\alpha_i y_i$；SVs 表示支持向量；nSV 表示正样本与负样本支持向量的数目。在得到 model 之后可以计算检测系统中的 Detector。其计算公式如下：

$$Detector=[sv\_coef'*SVs,-rho] \tag{3.10}$$

得到训练模型后可以读取测试数据进行预测，测试结果参见图 3.9。其正确率高达 100%，虽然如此这并不表示检测无误，在实际检测时需要在不同尺度上检测，可能出现误检。具体检测效果将在后文详述。

表 3.3 Matlab 的 SVM 训练代码

```
TrainPosFile='D:\VS2008\HogCV\HogCV\HogTrainP.txt';
TrainNegFile='D:\VS2008\HogCV\HogCV\HogTrainN.txt';
TrainP=textread(TrainPosFile);
TrainN=textread(TrainNegFile);
%构造训练数据TrainData以及样本对应的输出TrainLabel
TrainLabel=ones(800,1);
TrainLabel(401:800,:)=-TrainLabel(401:800,:);
TrainData=ones(800,10404);
TrainData(1:400,:)=TrainP(1:400,1:10404);
TrainData(401:800,:)=TrainN(1:400,1:10404);
model=svmtrain(TrainLabel,TrainData,'-s 0 -t 0 -c 1');
```

model <1x1 struct>

Field	Value	Min	Max
Parameters	[0;0;3;9.6117e-05;0]	0	3
nr_class	2	2	2
totalSV	102	102	102
rho	0.8714	0.8714	0.8714
Label	[1;-1]	-1	1
ProbA	[ ]		
ProbB	[ ]		
nSV	[47;55]	47	55
sv_coef	<102x1 double>	-0.0112	0.0095
SVs	<102x10404 double>	<Too many...	<Too...

图 3.8 SVM Model 即相关参数

```
Accuracy = 100% (200/200) (classification)
fx >>
```

图 3.9 SVM 测试结果

在得到 Detector 之后我们可以看看经过 Detector 加权后的 HOG 图，我们以图 3.7 中的正样本为例，进行比较，具体见图 3.10。

从图中可以看出再经过 Detector 加权后，其共性的部分，即图中红色部分更加突出显示。另外需要补充的是，这种表示方法可能不够准确，但在一定程度上还是说明了 SVM 的作用。

3.5 多尺度检测及 DetectSample 静态图像检测结果

对于实际检测手的大小肯定不是 144*144 大小，因而需要将对象进行缩放，达到在不同尺度上检测，从而顺利检测到手，这儿须注意两点：

- 尺度数
- 检测到多个目标区域是否为属于同一目标

对于尺度数目，采用表 3.4 所述策略：

表 3.4 确定尺度维数算法

```
Define：MaxLevels, Scale_Init
    Scale=Scale_Init
    For (Levels=0;Levels<MaxLevels;Levels++)
        If   检测窗口.width*Sclae>检测图像.width   or
             检测窗口.height*Sclae>检测图像.height
                    Break;
        Else
             Scale←Scale*Scale_Step;
    Levels=Max(Levels,1);
```

其中 MaxLevels 程序中取 24，Scale_Init 为 1，Scale_Step 为 1.05，每个尺度对应的尺度大小可记录在 Scale[Levels]数组中。

对于第二个问题，OpenCV2.0 已经为我们搭建好平台，可以直接调用函数：

void groupRectangles(vector<Rect>& rectList, int groupThreshold, double eps=0.2)

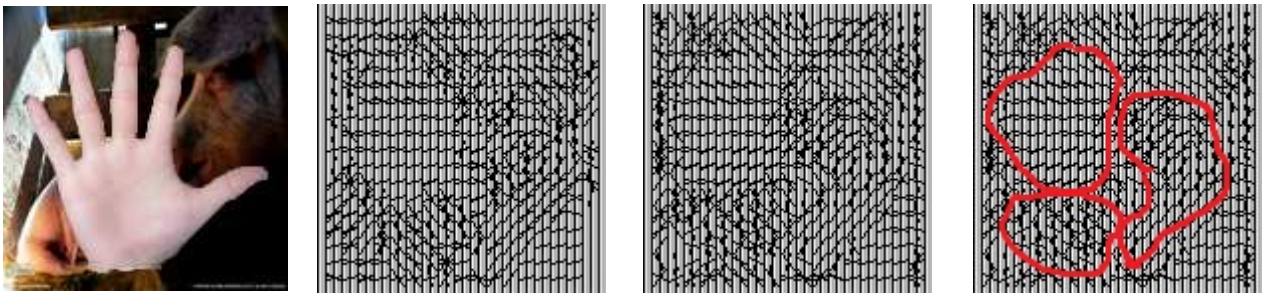
其中 recList 即为不同尺度上检测到手的矩形区域参数，groupThreshold 表示不同矩形区域属于同一个类别的相似度阈值，程序中取 2。检测效果可以见图 3.11~图 3.12。

对于 300 个检测样本：

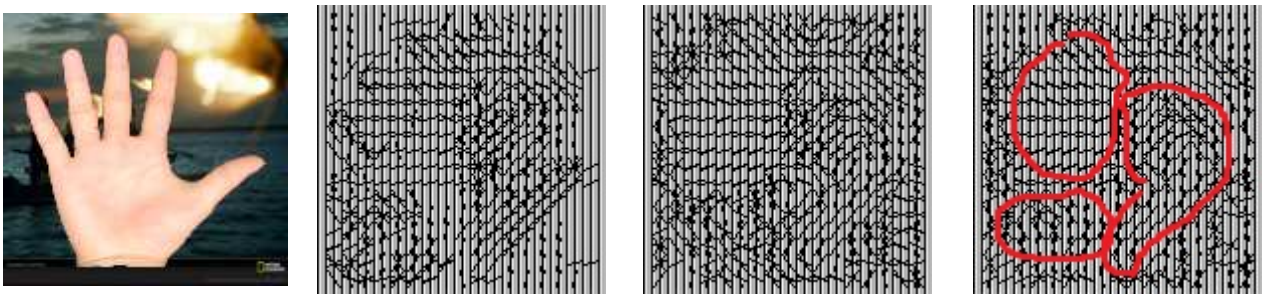
100 个友好正样本中，有 20 个样本出现了图 3.12 中 a、b 图所示误检测情况，未出现没有检测到手的情况。

100 个负样本中，仅有 2 个样本出现了图 3.12 中 c 图所示误检测情况。

100 个恶劣正样本中，有 35 个样本出现了图 3.12 中 a、b 图所示误检测情况，有 21 个样本出现图 3.12 中 e~f 所示的未检测到手的情况，由于手的旋转、张合变化过大。



(a)



(b)

图 3.10 从左到右分别为原始图像，HOG 图，经 Detector 加权后 HOG 图，用红色突出共同部分的经 Detector 加权后 HOG 图

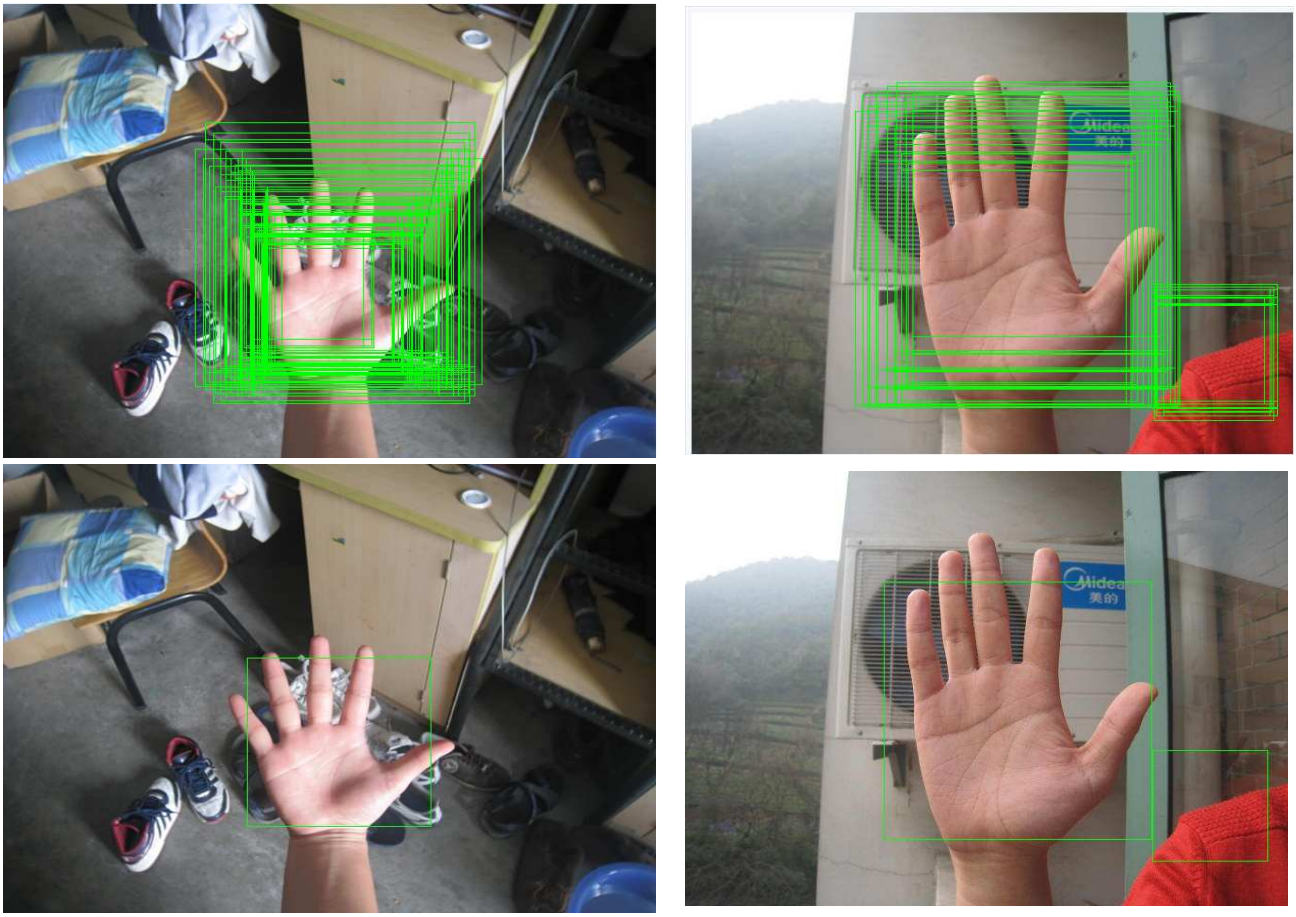
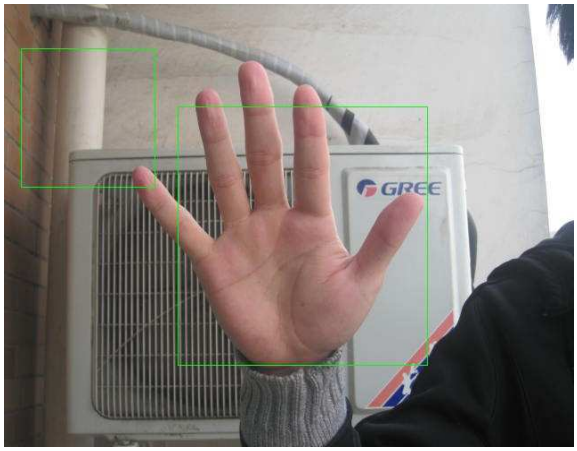
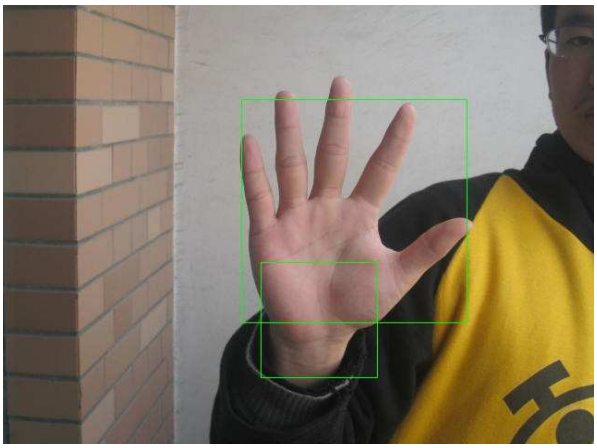


图 3.11 上层为未使用 groupRectangles 检测效果，下层为使用 groupRectangles 检测效果



(a)



(b)

(c)

(d)

(e)

(f)

图 3.12 静态图像检测中 a、b 为在较好正样本中出现误检；c 为在负样本中出现误检；d~f 为在较恶劣正样本中无法检测到手

3.6 动态图像检测结果

动态图像的检测与静态图像的检测区别在于能否达到实时性。经试验得到：

对于 640*480 的图像，用 1.73GHz 的处理器，耗时 12~13 秒；改用 2.53GHz 的处理器，耗时 5~6 秒。无法达到实时的效果。

如果缩小摄像头获取的图像大小，改为 320*240，用 2.53GHz 的处理器，可以将时间限制在 1 秒中以下，具有一定的实时性。

考虑到实验室服务机器人的处理器更加快速强大，对于 320*240 的摄像头输入图像是可能达到实时检测的，因而相当于完成了实时跟踪的目的。

动态图像的检测效果见图 3.13

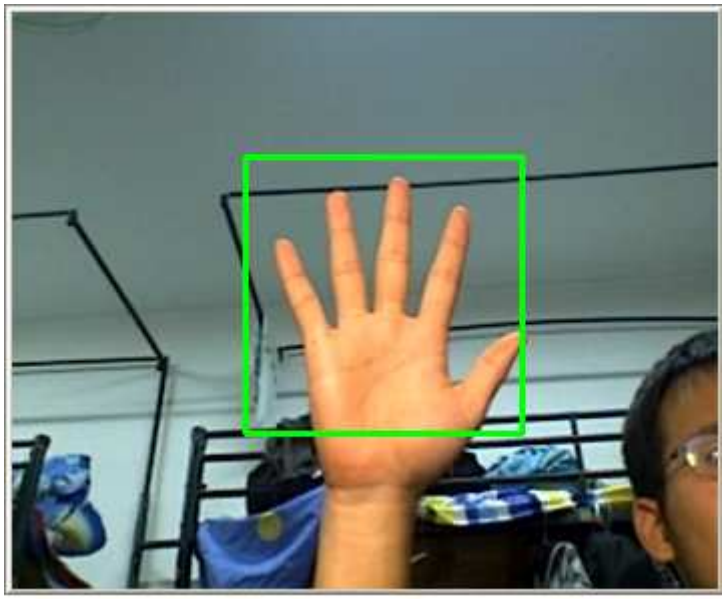


图 3.13 摄像头实际检测

3.7 基于检测前提下不同手势的识别

基于检测到张开状态下的手后再识别不同手势比直接识别手势简单许多。举个具体的例子，譬如我们可以得到不同手势对于 A 特征具有不同特征值，但我们无法仅仅根据 A 特征的不同值来推断当前对象是不同的手势，因为具有相同特征值的对象可以完全不是手。具体算法流程参见表 3.5。

- YCrCb 模型

其中 Y 是亮度，而 Cb 和 Cr 是色度信息。YCbCr 空间具有将色度与亮度分离的特点，

在 YCbCr 色彩空间中肤色的聚类特性比较好，而且是两维独立分布，能较好地限制肤色分布区域。

表 3.5 手势识别算法

Define: Detection_Frequency 检测频率
Detection：得到手的 RectArea，开始计时 Time
For: 当 Time<1/Detection_Frequency，Time++
Skin Model: 由 YCrCb 得到图像的二值图像，规定白色表示肤色
Draw Circle: 由 RectArea 确定圆心及半径
Classify：根据半径穿过的肤色区域数识别手势
If Time==1/Detection_Frequency，重新 Detection

YCbCr 可由 RGB 线性变化得到：

$$\begin{bmatrix} Y \\ Cr \\ Cb \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.1687 & -0.3313 & 0.5 \\ 0.5 & -0.4187 & -0.0183 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3.11}$$

虽然 YCrCb 模型能在一定程度上削弱 Y 亮度对颜色的影响，但简单的排除 Y 的影响将会削弱模型鲁棒性。因而需要对模型进行非线性分段色彩变换。由 $YCrCb \rightarrow Y'Cr'Cb'$，其转换公式如下：

$$Cb' = \begin{cases} [Cb(Y) - \overline{Cb(Y)}] \bullet \dfrac{W_{Cb}}{Wcb(Y)} + \overline{Cb(Y)}, & if \quad Y < Kl \quad or \quad Y > Kh \\ Cb(Y) \end{cases} \tag{3.12}$$

$$Cr' = \begin{cases} [Cr(Y) - \overline{Cr(Y)}] \bullet \dfrac{W_{Cr}}{Wcr(Y)} + \overline{Cr(Y)}, & if \quad Y < Kl \quad or \quad Y > Kh \\ Cr(Y) \end{cases} \tag{3.13}$$

其中：

$$Wcb(Y) = \begin{cases} WL_{Cb} + \dfrac{(Y - Y_{\min}) \bullet (W_{Cb} - WL_{Cb})}{K_l - K_{\min}}, & if \quad Y < K_l \\ WH_{Cb} + \dfrac{(Y_{\max} - Y) \bullet (W_{Cb} - WH_{Cb})}{K_{\max} - K_h}, & if \quad Y > K_h \end{cases} \tag{3.14}$$

$$Wcr(Y)=\begin{cases} WLcr+\dfrac{(Y-Y_{\min})\bullet(W_{Cr}-WL_{Cr})}{K_l-K_{\min}}, & if \quad Y<K_l \\ WH_{Cr}+\dfrac{(Y_{\max}-Y)\bullet(W_{Cr}-WH_{Cr})}{K_{\max}-K_h}, & if \quad Y>K_h \end{cases} \tag{3.15}$$

$$\overline{Cb(Y)}\begin{cases} 108+\dfrac{(K_l-Y)\bullet(118-108)}{K_l-Y_{\min}}, & if \quad Y<K_l \\ 108+\dfrac{(Y-K_h)\bullet(118-108)}{Y_{\max}-K_h}, & if \quad Y>K_h \end{cases} \tag{3.16}$$

$$\overline{Cr(Y)}\begin{cases} 154+\dfrac{(K_l-Y)\bullet(154-144)}{K_l-Y_{\min}}, & if \quad Y<K_l \\ 154+\dfrac{(Y-K_h)\bullet(154-132)}{Y_{\max}-K_h}, & if \quad Y>K_h \end{cases} \tag{3.17}$$

且有：$K_l=125, K_h=128$（非线性分段色彩变换的分段值）

$\mathrm{Y}_{\min}=16, Y_{\max}=235$（肤色聚类区域中 Y 分量的最小最大值）

$W_{Cb}=46.79, WL_{Cb}=23, WH_{Cb}=14$
$W_{Cr}=38.76, WL_{Cr}=20, WH_{Cr}=10$ （实验得到的常数）

经过这样的非线性分段色彩变换，我们得到肤色聚类在 YCrCb 空间中的分布集中在一个椭圆中。按照传统的方法，我们将一用一个椭圆来近似表示这个肤色区域，可得到如下公式：

$$\frac{(x-eCx)^2}{a^2}+\frac{(y-eCy)^2}{b^2}=1 \tag{3.18}$$

其中

$$\begin{bmatrix} x \\ y \end{bmatrix}=\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}\begin{bmatrix} Cb'-Cx \\ Cr'-Cy \end{bmatrix} \tag{3.19}$$

$Cx=109.38, Cy=152.02,$
$eCx=1.6, eCy=2.41, \theta=2.56$(弧度)
$a=25.39, b=14.03$

这样我们可以将肤色点与非肤色点进行分离，对图像进行二值化。在椭圆内的点设置为 255，不在椭圆内的点设置为 0。在实际处理中令

$$t=\frac{(x-eCx)^2}{a^2}+\frac{(y-eCy)^2}{b^2},\quad s=e^{-t} \tag{3.20}$$

若 $s<e^{-1}$，则认为是肤色点，否则不是。

➢ 画圆法识别

所谓画圆法是一种比较取巧，识别率不高，对背景和光照都有很高的要求。其实可将待识别的手势均通过 HOG+SVM 算法学习得到相应的 Detector，对于多值分类可以选择二叉树结构，将多个二值问题联合。但由于时间限制没有充分时间将每个手势都进行学习，因而采用一种取巧的方法。具体可参见图 3.14。

图 3.14 画圆法原理

所画的圆通过的肤色区域数目可与 1~5 的手势相对应。其中圆心和半径由检测环节中的矩形确定。具体代码中，所取圆心相对于矩形中心偏东南方向，半径为：

$$\alpha * \max(Rect.width, Rect.height) \tag{3.21}$$

式中 α 可调整，在 0.45~0.55 之间。

是否通过肤色区域的判断，是依据圆弧对应坐标的像素亮度值为 255（白色）的数目。当这个数目大于一个 Threshold 时则认为通过了肤色区域，代码中如下取值：

$$\beta * \max(Rect.width, Rect.height) \tag{3.22}$$

式中 β 可调整，在 0.03~0.07 之间。

需要补充的是为了避免手指之间的距离过于近，而将两不同肤色区域误认为同一肤色区域，在实际检测中对二值图像进行了腐蚀，使肤色区域变得更细，便于画圆法识别。

➢ 检测频率

之所以引进检测频率，是因为手势识别过程实际上分为两个步骤：1、检测 Open 状态的手；2、在检测到 Open 状态的手后，再识别此后变化的其他手势。由于摄像头实时识别时，手是会移动的，所以需要通过检测 Open 状态的手来不断更新手所在的矩形区域。而另一方面检测和识别有先后顺序关系，无法同时进行。

在代码中我们规定每 200 帧图片后检测手的最新位置。

识别效果参见图 3.15。


MainWindow
One
SkinWindow
One
MainWindow
Two
SkinWindow
Two
MainWindow
Three
SkinWindow
Three
MainWindow
Four
SkinWindow
Four

图 3.15 识别效果图

四、总结与展望

4.1 总结

首先，本文借助 Photoshop 制作了用于学习训练的手势数据库，其制作方法简单通用，可用于其它检测对象的数据库制作。其次，本次研究借助 OpenCV2.0 视觉库和 Visual Studio C++实现了基于 HOG 的特征提取和多尺度下检测模块；借助 LibSVM 和 Matlab 实现了基于 SVM 的学习分类模块。实现了在复杂背景下，对张开状态手进行多尺度的识别检测，实验数据统计显示命中率达 90%。最后，在检测到手的条件下，本文借助 YCrCb 肤色模型对不同手势进行识别。在背景与肤色高差别前提下，能基本实现对其他手势的识别。

在具体实现过程中如下几点非常重要：

数据库的制作：正所谓，工欲善其事，必先利其器。虽然样本的采集与制作非常耗费时间，但良好的训练数据，对基于学习的检测识别是至关重要的。另外，在制作数据库时，需要对样本的多样性与同一性做出权衡。

HOG 算法实现：HOG 算法中最核心的部分是直方图统计中的三次插值及高斯权重的应用，这部分内容貌似琐碎冗余，实际上是对局部信息的更加全面的放映。任何事物都是相互关联的，三插值法是从一个整体的角度去衡量局部的信息。而高斯权重则体现了对事物的评估，不同组成成分对事物的贡献程度是不一样的。

SVM 模块应用：在运用 SVM 学习分类时，常选择的核函数是径向基核，但在本文中我们优先线性核。其原因是由线性核学习得到的分类器，可以通过公式计算出 HOG 检测模块中的 Detector 算子，将其作为常量保存调用。若用径向基核，则对于每个样本计算得到 HOG 特征向量后，还需要调用 SVM 的预测函数，更为复杂耗时。

识别模块：在得到对象的 YCrCb 图像后，对图下进行腐蚀操作，可以减少非手的类肤色区域，更有利于识别。

4.2 展望

本次研究，不足之处有以下几点，有待日后改进完善：

- 在复杂背景下，对张开状态手的检测虽然有高达 90%的命中率，但是也存在近 20%误识率，需要对算法进一步改善
- 对于手的检测算法过复杂耗时，一张 640*480 的图片，基于 1.73GHz 的处理器需要近 13 秒的时间才能检测到，这对于实时监测是致命的。因而算法还需要进一步优化。
- 对于其他手势的识别是简单的基于肤色模型。而肤色模型对于背景、光照有较高的要求。尤其当背景中出现了与肤色类似的物体时，其识别错误率骤升。需要考虑更鲁棒性的算法。

参考文献

[1] 王金庭, 杨敏. 基于 YCbCr 空间的亮度自适应肤色检测. 计算机系统应用, 2007 年,第 6 期:99~102

[2] Paul V, Micheal J. Robust Real-time Object Detection. Cambridge Research Laboratory Technical Report Series, 2001,01: 1~23

[3] Paul V, Micheal J. Rapid Object Detection Using a Boosted Cascade of Simple Features. IEEE International Conference on Computer Vision and Pattern Recognition, 2001, vol.1: 511~518

[4] Rainer L, Jochen M. An Extended Set of Haar_like Features for Rapid Object Detection. IEEE International Conference on Image Processing, 2002,vol.1: 900~903

[5] David G. Object Recognition from Local Scale-Invariant Features. Proceedings of the IEEE International Conference on Computer Vision, 1999, vol.2: 1150~1159

[6] David G. Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, 2004,11, vol.60, no.2: 91~100

[7] Tony L. Scale-space theory: A basic tool for analyzing structures at different scales. Journal of Applied Satistics, 1994, vol.21, no.2: 225~270

[8] 宋丹. SIFT 算法详解及应用. http://wenku.baidu.com/view/20dfb0671ed9ad51f01df2ee.html

[9] Charles Z, Ralph R. Fourier Descriptors for Plane Closed Curves. IEEE Transactions on Computers, 1972, 05, vol.21, no.3: 269~281

[10] Martin L, Jan V, Joachim B. Distortion Invariant Object Recognition in the Dynamic Link Architecture. IEEE Transactions On Computer, 1993,05, vol. 42, no.3:300~311

[11] Laurenz W, Jean-Marc F, Norbert K. Face Recognition and Gender Determination. 1995

[12] Rainer L, Alexander K, Vadim P. Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection. Pattern Recognition Symposium, 2003, vol.25:297~304

[13] 吴暾华, 周昌乐. 快速人脸检测系统的设计与实现. 计算机应用, 2005,10, vol.25, no.10: 2350~2353

[14] SVM 教程. http://wenku.baidu.com/view/62b93921dd36a32d73758102.html

[15] Martin H, Howard D, Mark B 著. 《神经网络设计》. 戴葵 译. 北京:机械工业出版社, 2002

[16] 栾丽华, 吉根林. 决策树分类研究. 计算机工程, 2004, 05,vol.30, no.9: 94~96

[17] 周航. 基于计算机视觉的手势识别系统研究. [博士论文]. 北京:北京交通大学. 2007

[18] Navneet D, Bill T. Histograms of Oriented Gradients for Human Detection. IEEE International Conference on Computer Vision and Pattern Recognition, 2005, vol.1: 886~893

[19] Navneet D. Finding People In Images and Viedeo. [PhD Thesis]. France: French National Institute for Research in Computer Science and Control

[20] Nello C, John T 著. 《支持向量机导论》. 李国正,王猛,曾华军 译. 北京：电子工业出版社. 2004