

云南大学

本科实验报告

课程名称: 图像理解与计算机视觉

实验名称: 实验七. 计算机视觉典型应用实验

学院（系）：_____

专 业: _____

班 级: _____

姓 名: _____

学 号: _____

指导教师: _____

成绩: _____

评语:

Steven

一. 实验目的

学生通过查阅资料掌握1-2种给定计算机视觉典型应用的基本原理，能够针对具体应用复现1-2种经典算法，并通过调参等手段对具体算法调优。

二. 实验内容

(1) 采用三帧差法实现给定视频序列的运动目标检测。



(2) 采用高斯混合背景建模法实现(1)中给定视频序列的运动目标检测，并分析高斯分布的数量、判别是否符合一个高斯分布的阈值、背景模型学习率三个参数对检测结果的影响情况。

(3) 根据给定视频序列的第一帧目标位置和尺寸，采用核相关滤波器算法(KCF)在后续帧序列中实现运动目标跟踪，并对比分析Gray特征、HOG特征的跟踪效果。



三. 实验环境

Matlab软件是图像处理领域广泛使用的仿真软件之一。本实验基于Matlab 2022版本完成。

四. 实验代码（只贴核心代码段，详细注释，Times New Roman/宋体 五号字体 单倍行距）

```
%% 使用: 在命令行内调用函数 Exp7(func), func 是不同功能的名字。  
%% 每题的 demo 调用格式如下:
```

```

% 1. 三帧差法 运动目标检测: Exp7("Ques1");
% 2. 高斯混合背景建模法 运动目标检测: Exp7("Ques2");
% 3. KCF 运动目标追踪: Exp7("Ques3");
% ---要先修改/tracker/run_tracker.m'中第 41 行的'base_path'为 Girl2 的上级文件夹;
function Exp7(ques)
    if ques == "Ques1"
        video = VideoReader('pedestrian.avi');
        frame_num = get(video, 'NumFrames');
        frame_difference_3(video, frame_num);
    elseif ques == "Ques2"
        video = VideoReader('pedestrian.avi');
        gauss_bg_modeling(video);
    elseif ques == "Ques3"
        cd tracker\
        run_tracker('choose','gaussian','hog',~strcmp('choose', 'all'),~strcmp('choose', 'all'));
        run_tracker('choose','gaussian','gray',~strcmp('choose', 'all'),~strcmp('choose', 'all'));
    end
end

function frame_difference_3(video, frame_num)
    [width, height, ~] = size(read(video, 1));
    for i = 1:frame_num - 3
        frame1 = read(video, i);
        frame2 = read(video, i + 1);
        frame3 = read(video, i + 2);
        difference1 = abs(rgb2gray(frame1) - rgb2gray(frame2));
        difference2 = abs(rgb2gray(frame2) - rgb2gray(frame3));
        difference1 = imbinarize(difference1, 0.2);
        difference2 = imbinarize(difference2, 0.2);
        difference = difference1 & difference2;
        imshow(frame2);
        hold on
        for i = 1:width
            for j = 1:height
                if difference(i, j) == 1
                    plot(j, i, 'r.');
                end
            end
        end
        hold off
        pause(0.1);
    end
end

```

```

function gauss_bg_modeling(video)
    %-----混合高斯背景建模 参数 -----
    gauss_n = 3; %每个像素点高斯背景模型数量
    a = 0.005; %学习速率    alpha
    vt = 2.5^2; %方差阈值    2.5*2.5 倍的方差 VarThreshold
    bgr = 0.7; %背景比率    BackgroundRatio
    w0 = 0.05; %初始权值    weight
    var0 = 10^2; %初始方差    variance

    %-----混合高斯背景建模 读取视频参数-----
    f_n = video.NumberOfFrames; %帧数 frame_num
    f = rgb2gray(read(video, 1)); %读取第一帧灰度图像
    height = video.Height; %获取图像的高度
    width = video.Width; %获取图像的宽度

    %-----初始化高斯背景模型 共有 height*width*gauss_n*3 个数值-
    %每一个像素对应 gauss_n 个高斯背景模型 每个模型有三个参数[权值 均值 方
    差]
    g_b = zeros(height, width, gauss_n, 3);

    for h = 1:height
        for w = 1:width %像素遍历
            g_b(h, w, 1, 1) = 1; %第一个模型初始权值为 1
            g_b(h, w, 1, 2) = double(f(h, w)); %第一个模型初始均值为第一帧灰度图像素
            点的值
            g_b(h, w, 1, 3) = 9; %初始方差
        end
    end %此方式初始化容易将第一帧内的运动物体也当成背景 最好使用前 n 个帧训练
    模型 or 一开始的学习率很高

    %-----进行匹配 更新模型-----
    %帧遍历
    for n = 2:f_n
        f = rgb2gray(read(video, n)); %读取下一帧
        %像素遍历
        for h = 1:height
            for w = 1:width
                khit = 0; %匹配的模型序号 默认与第一个模型匹配
                bg_n = 0; %描述背景的高斯模型数量
                %高斯模型遍历
                for k = 1:gauss_n
                    ww = g_b(h, w, k, 1); %模型权值
                    if (ww == 0) %权值为 0 则模型为空 跳过

```

```

        continue;
    end
    mean = g_b(h, w, k, 2); %模型均值
    var = g_b(h, w, k, 3); %模型方差
    diff = double(f(h, w)) - mean; %像素点与模型均值的差
    d2 = diff^2; %差的平方
    %与此模型匹配成功
    if (d2 < vt * var)
        g_b(h, w, k, 1) = ww + a * (1 - ww); %增加权值
        g_b(h, w, k, 2) = mean + a * diff; %更新均值
        g_b(h, w, k, 3) = var + a * (d2 - var); %更新方差
        khit = k; %记录匹配的模型序号
        %模型排序 从后向前冒泡
        for kk = k:-1:2
            ww1 = g_b(h, w, kk, 1); %权值
            var1 = g_b(h, w, kk, 3); %方差
            ww = g_b(h, w, kk - 1, 1); %权值
            var = g_b(h, w, kk - 1, 3); %方差
            %大于前一个 则交换
            if (ww1 / sqrt(var1) > ww / sqrt(var))
                tmp = g_b(h, w, kk, :);
                g_b(h, w, kk, :) = g_b(h, w, kk - 1, :);
                g_b(h, w, kk - 1, :) = tmp;
                khit = khit - 1; %匹配的模型序号更新
            end
        end
        break;
    end
end

%全部匹配失败 新建立模型覆盖权值为 0 or 最后一个模型
if (khit == 0)
    for k = 2:gauss_n
        if (g_b(h, w, k, 1) == 0 || k == gauss_n)
            g_b(h, w, k, 1) = w0;
            g_b(h, w, k, 2) = double(f(h, w));
            g_b(h, w, k, 3) = var0;
            break;
        end
    end
    khit = k; %匹配的模型序号变更
end

%权值归一化 保证权值和为 1

```

述背景

```
wsum = sum(g_b(h, w, :, 1));
bt = 0;
for k = 1:gauss_n
    %%%
    g_b(h, w, k, 1) = g_b(h, w, k, 1) / wsum;
    bt = bt + g_b(h, w, k, 1);
    %前 bg_n 个模型的权值和 大于背景比率 则前 gb_n 个模型来描

    if (bt > bgr && bg_n == 0)
        bg_n = k;
    end
end

%二值化
if (khit > bg_n) %匹配的模型 不是前 gb_n 描述背景的模型
    f(h, w) = 255;
else %匹配的模型 属于用来描述背景的模型
    f(h, w) = 0;
end
end
end
clc;
fprintf('进度: %d / %d \n', n, f_n);
imshow(f);
end
disp('OK!');
end
```



Exp7.m



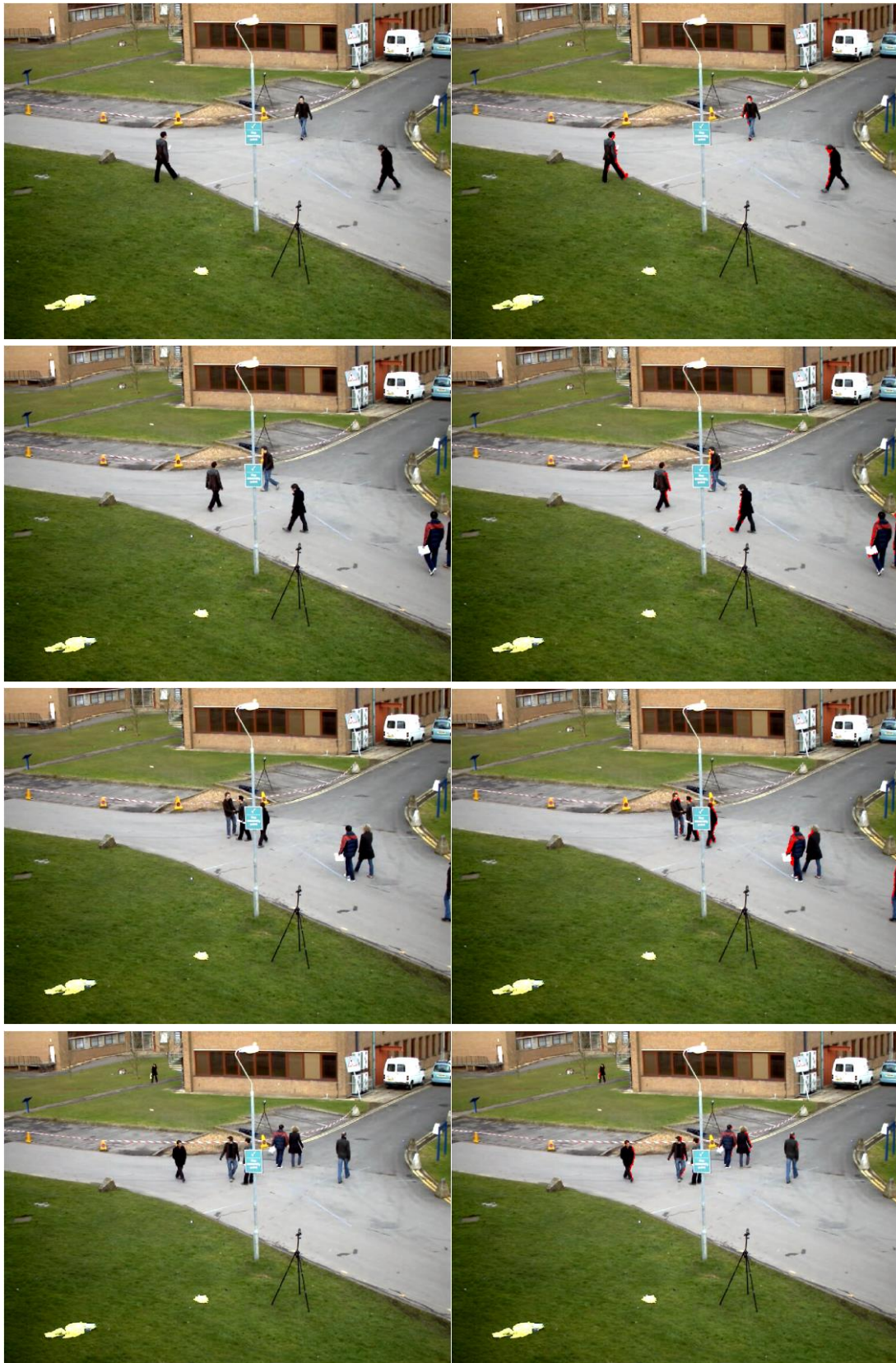
tracker.zip

附件（.m文件）：

五. 实验结果及分析

（1）三帧差法检测结果展示（给出3-5帧原视频帧和对应检测结果）；

以下是在debug模式下获取的原图以及带检测标记的结果。



（对于判定为移动目标的区域，使用红点进行描边）

（2）高斯混合背景建模法检测结果展示，对于每一种特定的高斯分布数量、判别是否符合一个高斯分布的阈值、背景模型学习率三个参数，分别给出3-5

帧检测结果，并分析三个参数对检测效果的影响；



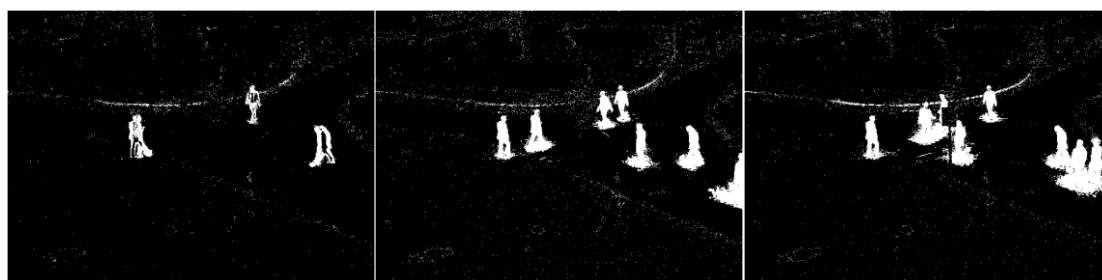
（高斯分布数=3; 阈值= $2.5^{2.5}$ 学习率=0.005; 选取2、12、22帧）



（高斯分布数=10; 阈值= $2.5^{2.5}$ 学习率=0.005; 选取2、12、22帧）



（高斯分布数=3; 阈值= $1.5^{1.5}$ 学习率=0.005; 选取2、12、22帧）



（高斯分布数=3; 阈值= $2.5^{2.5}$ 学习率=0.01; 选取2、12、22帧）

分析上述几种情况可知：

高斯分布数对检测效果影响较小；

阈值越高，检测越精确，但阈值过高则容易使得检测结果有缺失。

学习率越高，检测越细致，但同时也容易出现范围缺失的问题。

（3）对于Gray特征、HOG特征分别给出KCF算法3-5帧代表性的跟踪结果，并分析两种特征的优劣。

(1) Gray特征



(2) HOG特征



两种特征下的KCF算法都出现了目标丢失的现象，HOG出现目标丢失的更早，而Gray的定位框最终超出了视频范围，二者的效果都并不太理想。

六. 实验体会