目标分类

读取目标图片

中值滤波

trainAndtest

目标特征提取

预测

读取文件并进行特征提取readFolderAndExtractFeatures

合并数据trainingDataMat

创建模型svm = SVM::create();

开始训练svm->train(trainingDataMat, ROW\_SAMPLE, responses);

开始预测svm->predict(testDataMat, testPredict);

错误估计 error= 100.0f \* countNonZero(errorMat) / testResponsesData.size();

这个应该是标记plotTrainData(trainingDataMat, responses, &error);

Mat pre= preprocessImage(img);

features= ExtractFeatures(pre, &pos\_left, &pos\_top);

Mat trainingDataMat(1, 2, CV\_32FC1, &features[i][0]);

result= svm->predict(trainingDataMat);

images.read(frame)

Mat pre= preprocessImage(frame);

vector< vector<float> > features= ExtractFeatures(pre);

trainingData.push\_back(features[i][0]);

trainingData.push\_back(features[i][1]);

responsesData.push\_back(label);

testData.push\_back(features[i][0]);

testData.push\_back(features[i][1]);

testResponsesData.push\_back((float)label);

转化成灰度图片

中值滤波

去掉背景

二值化

findContours(input, contours, hierarchy, RETR\_CCOMP, CHAIN\_APPROX\_SIMPLE);

Mat mask= Mat::zeros(img.rows, img.cols, CV\_8UC1);

drawContours(mask, contours, i, Scalar(1), FILLED, LINE\_8, hierarchy, 1);

Scalar area\_s= sum(mask);

float area= area\_s[0];

一个是面积 一个是比列两个特征

if(area>500)

row.push\_back(area);

row.push\_back(ar);

output.push\_back(row);