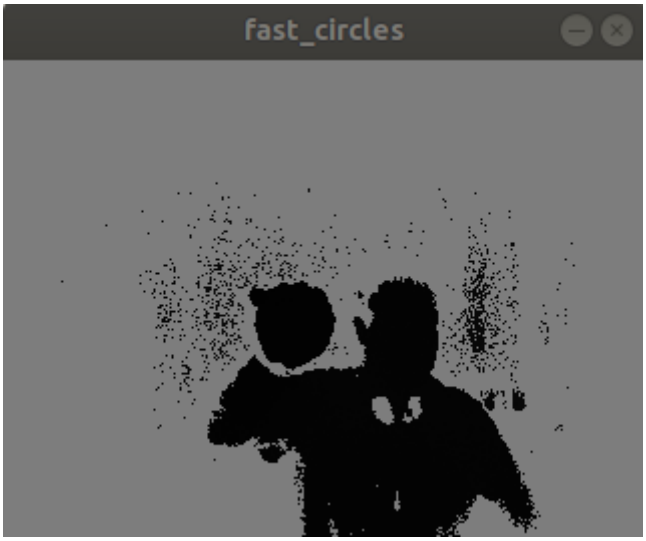
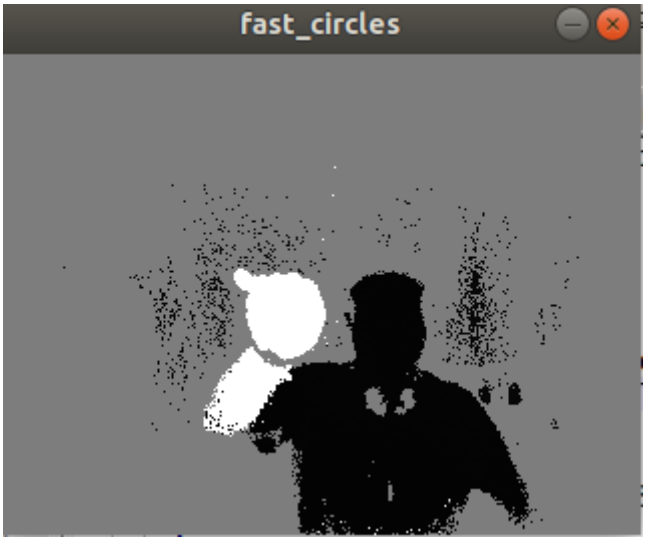
Interaction Between Classes And Algorithm Overview

1. **Class MultiStrokeRec.cpp:** Main Class in order to Run the Project
2. **DirAccess.cpp:** class to load the and simulate a video stream in the application



3. **ImageProcessing.cpp:** The core of the project

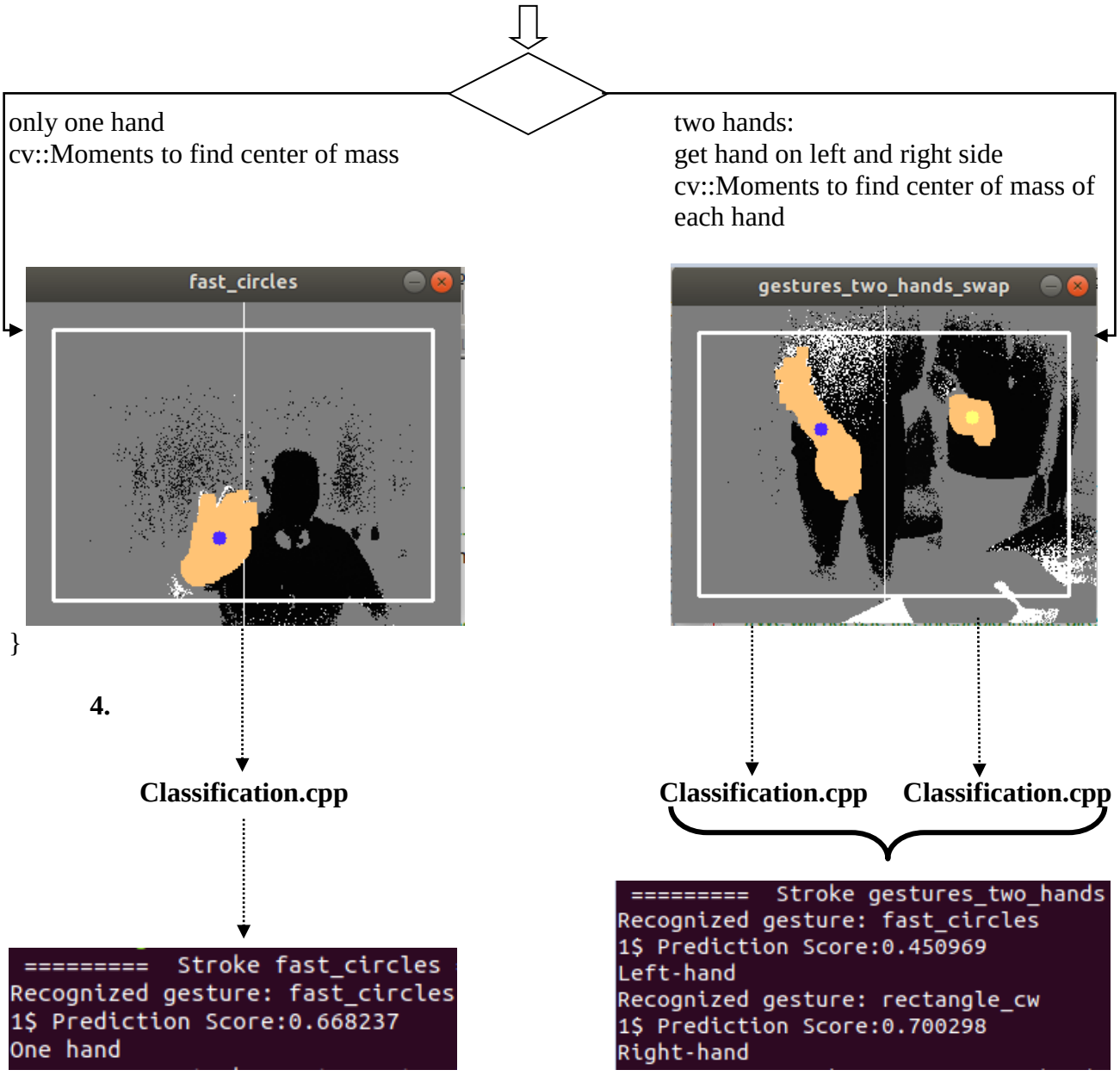
```
//Calculates optimized threshold using histogram equalization by iterations
void ImageProc::HistEqu(cv::Mat* ImgSrc, int32_t* Thres, uint16_t bins)
{
  this->Hist(ImgSrc, &Equhist, bins)              //Calculate histogram
  this->CumHist(&Equhist, &EquCumHist)      //Calculate cumulative histogram
  ...
}

void ImageProc::Thres(cv::Mat& ImgSrc, cv::Mat& ImgDest, int32_t* Thres)
{
    //Segment the hand based on a threshold
```




```
}
```

//Calculates the center of the mass utilizing OpenCV available resources
void ImageProc::BlobAnalysis(cv::Mat& ImgSrc, cv::Mat& ImgDest)
{

a) Calculate the ROI
b) Divide the screen in two vertical parts
c) cv::inRange detects object based on range of pixel values that fall under the threshold
d) cv::erode and cv::dilate to clean image and remove false positives. It also makes the threshold object more visible
e) cv::findContours to join continuous points with same intensity
f) get the biggest contour and the second biggest (if exists)
g) Verify how many hands were found:



Classification.cpp uses the sequence of positions to classify the stroke with $1 Unistroke Recognizer.

Implementation Guide

Dependencies

OpenCV https://docs.opencv.org/3.3.1/d7/d9f/tutorial_linux_install.html

Go to the project dir

open CMakeLists.txt and adjust the path to your OpenCV dir

in command line cd **<root project dir>**

Build

cmake CMakeLists.txt

make

Run

./bin/MultistrokeRec