

Individual Lab Report

Name: Sida Wang

Team letter and name: Team G, RoboGrapher

Teammates: Sida Wang

Tiffany May

Gauri Gandhi

Jimit Gandhi

Rohit Dashrithi

ILR Number: ILR 02

Submission date: Oct. 23, 2015

1. Individual progress

(1). Algorithms for human detection

I mainly worked with Tiffany to focus on human detection. We aim to find human beings in the figures. We used HOG (histogram of oriented gradients) algorithm to solve this problem. HOG is the most popular algorithm for human detection. Here are the basic steps:

(a). gradient computation

The first step of calculation is the computation of the gradient values. And the most common method is to apply derivative mask in horizontal and vertical directions with the filter kernels: $[-1 \ 0 \ 1]$ and $[-1 \ 0 \ 1]^T$.

(b). Orientation binning

In order to compute faster, the image should be divided into several cells or blocks, and each cell consists of 6 or 8 pixels, and each block consists of 9 cells, and we would calculate the overall gradient for each block. And the histogram are evenly spread over 0 to 180 degrees (20 degrees each, and 9 bins in total).

(c). Descriptor blocks

The gradients should be locally normalized.

(d). SVM classifier

We would use SVM (support vector machine) classifier is a binary classifier to classify images.

(2). Human detection software implementation

I and Tiffany implemented HOG human detection in MATLAB. The reason why we used MATLAB instead of C++ is that MATLAB is more easier and faster to implement this algorithm, and MATLAB is easier for us to test the algorithm. We used an image including 12 people standing or sitting in the image. And the result is shown in Drawing 1.

(3). Result analysis

We used the parameter of missing rate to quantitatively analysis the algorithm of HOG for human detection. And HOG is mainly used for detecting standing people.

Missing rate for standing people: $6 / 8 = 75\%$

Missing rate for all the people: $6 / 12 = 50\%$

The running time for HOG algorithm in this case is 0.33s.

We think that the accuracy is acceptable, but the running time is slow. As we should implement vivo human detection in our case, the time 0.33s is too long to tolerate.



Drawing 1: The result of human detection using HOG algorithm

2. Challenges

(1) The speed of HOG algorithm in MATLAB

We are thinking about using C++ to detect human-beings to make it faster.

(2) IntraFace

We still have some problems with IntraFace software.

As the IntraFace has just released a new version, there is still something wrong with it, and we are contacting the IntraFace people to solve it.

home/tiffany/Documents/API/1009Deliverable_x64/demo_source_release/include/intraface/emo/EmoDet.h:28:54: sorry, unimplemented: non-static data member initializers

/

home/tiffany/Documents/API/1009Deliverable_x64/demo_source_release/include/intraface/emo/EmoDet.h:28:54: error: 'constexpr' needed for in-class initialization of static data member 'emoNeutral' of non-integral type

.....

3. Teamwork

This week, I and Tiffany mainly worked on human detection, and tried to learn IntraFace. Rohit mainly worked on pan-tilt part. And Jimit and Gauri prepared for multi-camera face recognition and navigation.

4. Future work

Next week, I and Tiffany plan to use IntraFace after the bug has been fixed. Furthermore, we will use C++ to implement HOG algorithm. Wish things will become better and better! Happy the next half semester!!