Street Lane Recognition with OpenCV

Mike Facelle

OpenCV

Open-source “Computer Vision” API

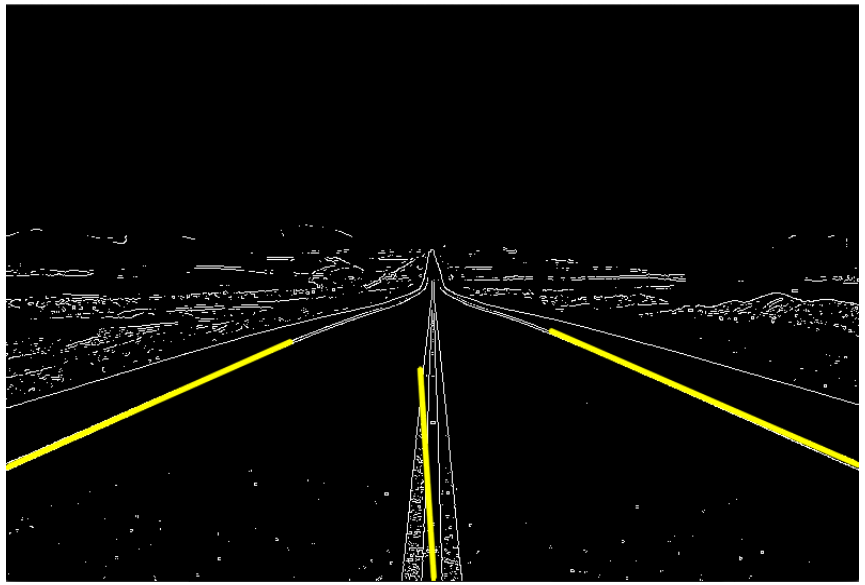
Java, Python, C/C++, etc

Using C++ with Xcode IDE

Objective

Mark lanes on a road

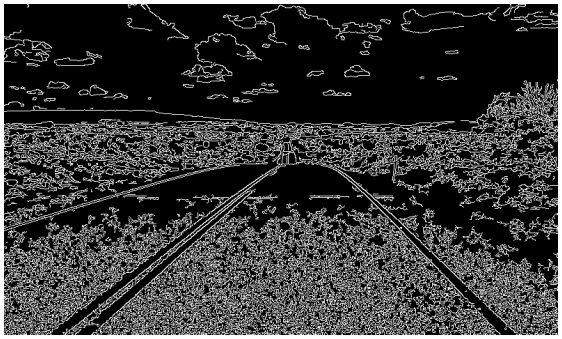
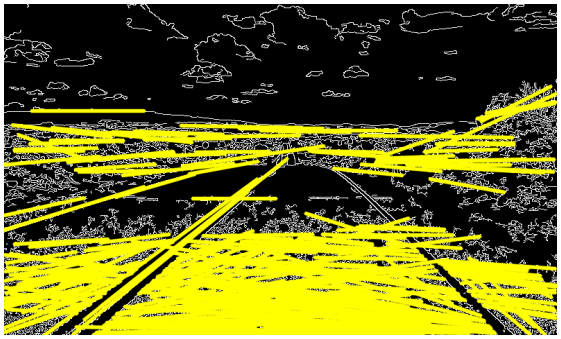
Maybe mark which lane photo is taken in



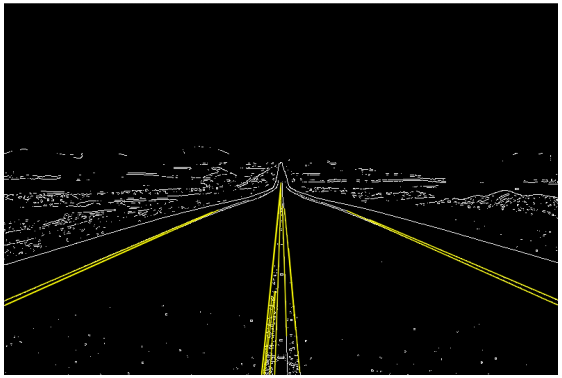
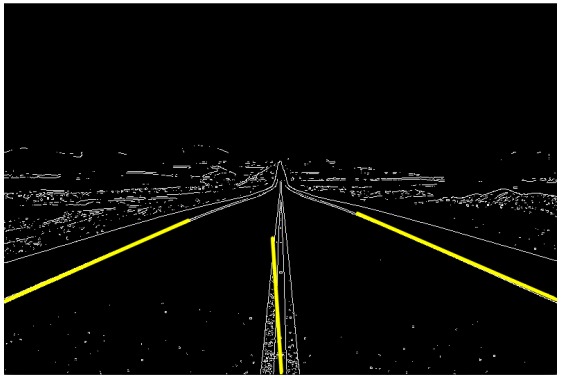
(Current Progress)

Problems

Requires a clear, good-quality image

Too many lines drawn - must combine them

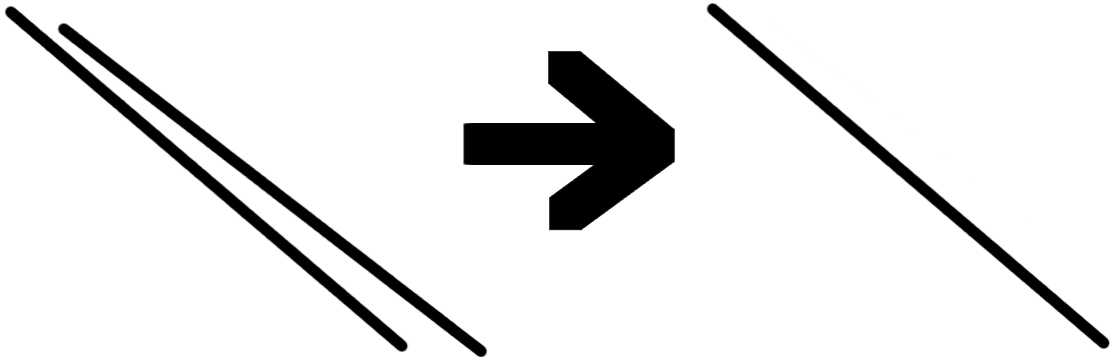




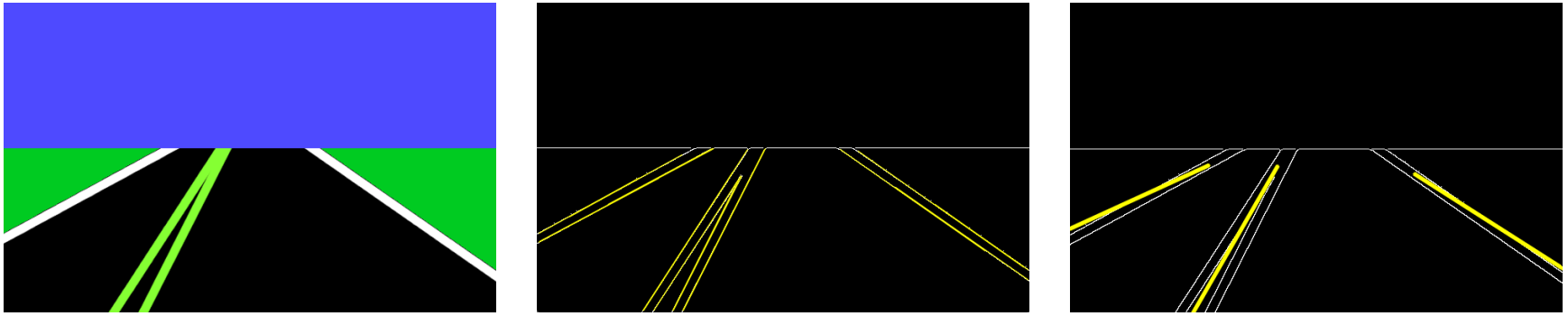
Fixing Problems - project.h

```
// functions to reduce number of lines in image
// ---
void remove_horizontal(vector<Vec4i> *);        // removes horizontal lines from the vector<Vec4i>
void remove_lines(int,int,vector<Vec4i>*);      // removes two lines from a vector of lines
bool horizontal(Vec4i);                         // determines if a line is horizontal (below tolerance)
// ---
vector<Vec4i> combine_lines(vector<Vec4i>);        // combines adjacent/seperated lines
vector<Vec4i> extend_lines(vector<Vec4i>,int,int);  // extends lines to reach end (bottom, edges) of
screen
// ---
bool greater_than(Vec4i, Vec4i);          // returns true if first line is higher up than second
void swap(Vec4i*, Vec4i*);                // swaps two lines to pass to adjacent/seperated() correctly
// ---
bool same_line(Vec4i, Vec4i);             // returns true if l1,l2 are the "same" line
bool adjacent(Vec4i, Vec4i);              // returns true if l1,l2 are adjacent
bool seperated(Vec4i, Vec4i);             // returns true if l1,l2 are seperated but the "same" line
// ---
double slope(Vec4i);                      // returns slope of the line passed
double y_intercept(Vec4i);                // determine y-intercept of line passed
double x_intercept(Vec4i);                // determine x-intercept of line passed
int mean(int,int);
```

Fixing Problems - Methods

Two lines are “the same” if their slopes are equal and their x-intercepts are equal (within a tolerance).





Still to do

Must fill in the space between lines to designate the lanes

Test with more images

Implement on Raspberry Pi (ideally using Makefiles)