TJHSST Computer Systems Lab 2007 - 2008
Car Recognition Through SIFT Keypoint Matching
by Drew Stebbins

Abstract

Many law enforcement agencies have recently shown interest in automated automobile recognition and tracking technologies such as license plate recognition (LPR) or GPS tracking. However, some criminals may drive vehicles that have false license plates or are not equipped with GPS tracking devices, making the pursuit of such vehicles difficult or impossible. This project aims to create a computer vision system capable of taking real-time input from a static camera and identifying passing cars by make and model in order to assist law enforcement agencies in the tracking of suspect or stolen vehicles.

Background

Much research has been done on the topic of generalized computer-based object recognition through the use of techniques such as the Viola-Jones face recognition algorithm and David Lowe's Scale Invariant Feature Transform (SIFT) keypoint matching method. Dlagnekov and Belongie, at the University of California, San Diego, have made an attempt at a car recognition system based off of region of interest (ROI) identification through the use of LPR and actual object recognition through the use of SIFT keypoint matching. They recorded a successful recognition rate of 89.5% on their query set of 38 images using a database of 790 images of known make and model.

Procedure and Methods

My program processes images of cars taken in side profile, unlike the rear angle from which all images used by Dlagnekov and Belongie were captured. Currently, my program can determine the proper ROI in an image by locating a car's wheels through the use of the Hough circle transform, and locate all SIFT keypoint candidates through the use of Gaussian smoothing and the difference-of-Gaussian function. The Hough transform-based circle detector makes multiple passes through Hough space for each possible car wheel radius, searching for the highest Hough space intensity values in each side of the image. Scale space extrema are detected by comparing the difference in the value of a pixel at one scale to the value of the pixel in the same place at another scale to that same value at all of the locations adjacent to that pixel in all current and neighboring scales. Once my program is able to locate and describe SIFT keypoints, it will make a best guess as to the make and model of the car in the input image based on the number of matching SIFT keypoints between the input image and each image in the database of cars of known make and model.

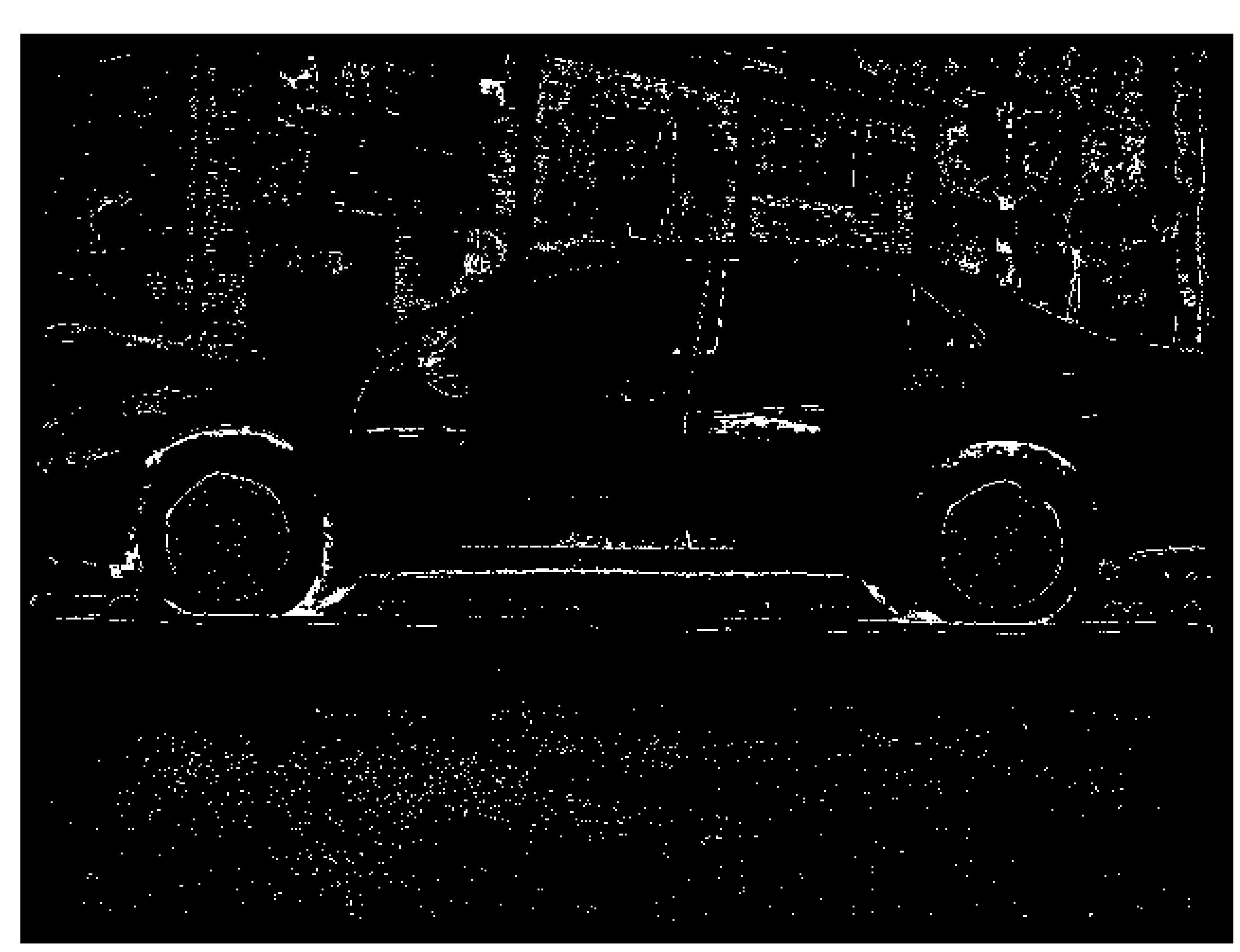
I have also developed a separate program to receive and process real-time input from a USB webcam attached to a Linux computer. Once I have completed my implementation of the SIFT keypoint matching algorithm, I intend to combine it with this program to produce a program with a GUI that allows the user to freeze any given frame and detect the model and make of each car located within the camera's view at that particular moment. If I have time, I also intend to combine motion detection with wheel detection to enhance the accuracy of my ROI detection algorithm.

Results

My ROI detection algorithm currently performs very well on images of cars with wheels of any size, as long as the amount of random noise is kept below a certain threshold. Above a certain noise level, the extra edge pixels detected by the Canny edge detector make accurate wheel detection nearly impossible. My implementation of the SIFT keypoint matching algorithm is currently roughly 20% completed, and once it is finished, I intend to test it on a database of side-profile car image I have collected from the Thomas Jefferson student parking lot. I hope to achieve a recognition rate similar to that recorded by Dlagnekov and Belongie.

Sample testing image

Detected ROI (based on position of car wheels)



Scale-space extrema (SIFT keypoint candidates)