Project Report

Car Logo Detection and Recognition

Pratik Bongale, M.S., Computer Science Dept., Rochester Institute of Technology

Abstract:

Vehicle Manufacturer Recognition(VMR) is a popular vision-based system widely used in secure access, vehicle verification and traffic monitoring applications. This paper proposes an investigation into recognizing the make of a car from its brand logo. Recognizing an automobile's make is a challenging task due to lack of discriminative features and close appearance of automobile logos. This paper presents a relatively simple algorithm using cross-correlation to detect the logo location and recognize the make of car in the given image. This algorithm is assessed on a set of 240 test images and has resulted in promising accuracy for logo detection as well as recognition.

Introduction:

Car-logo is a key information for make recognition. The recognition problem is a simple task of comparison; however, an essential and challenging task is to determine the location of logo in an arbitrary image which can be fed to the recognition module. Thus, the make recognition process is composed of two modules, firstly logo detection module – to accurately determine the location of logo in the query image and secondly logo recognition module - to determine the make of car by comparing the detected logo with a dataset of logos.

Intuitively, one would think of using edge detection to find regions in the image which resemble a pre-computed template of logo. In natural images, a car can be at any arbitrary distance from the camera, we need to accommodate various scales of logos to tackle this problem. Car-logo is a small object as compared to other objects in an image, any edge detection technique will output numerous edges from other objects which are noise in our application. Most of car-logos have the same shiny silver color, so classification using color as a feature fails to solve our purpose as we are unable to distinguish between two logos. Using characteristic features (keypoints) generated by corner detectors like Harris corner detector or SIFT features produces large number of features in the image, but the logo region being small, these corner detectors fail to generate enough keypoints in the logo region to be able to recognize the logo.

This paper presents an effective and efficient algorithm for make recognition using template matching which addresses many of the issues in logo detection phase mentioned above. To accommodate different scales of logo in natural images, we build a scale space of a logo template. We convert the input image to gray scale as we will not be using the color information in this algorithm and having to deal with only grayscale information will increase processing speed. We calculate cross-correlation between the query image and logo templates from scale space and use the correlation co-efficient of the match between query image and template as a measure of degree of confidence to determine the make of the car in query image.

Previous Work:

The small size of logo and lack of discriminative features is a major challenge in the initial step of logo detection. A common approach is to convert the original problem of logo detection into an easier problem of locating the surrounding objects near logo. Different algorithms use different objects such as license plate, headlights, radiator grille, wind shield and so on to obtain a coarse location of logo in given image. Yang Lui and Shutao Li [1] and Yunqiong Wang, Zhifang Liu Fei Xiao [2] uses the license plate location information to restrict their search for logo only in the region above the license plate. This approach only performs logo detection in images with frontal view of car, because in some vehicles the location of logo on the rear end is not always relative to the license plate location. Thus, using contextual information for logo detection fails for rear view images.

Another common approach is to extract the unique characteristic features in the image (keypoints) and compare them with the logo features stored in a dataset of logos. Apostolos P. Psyllos , Christos-Nikolaos E.

Anagnostopoulos , Eleftherios Kayafas [3] have proposed an algorithm to recognize logos by comparing the SIFT features from a small region above the license plate with their logo features dataset. They enhanced the original SIFT[4] algorithm proposed by David Lowe to fit the purpose of this application. Using descriptive features works great in some instances as they look for scale and rotation invariant features in the image. However, they do not produce good results when there are illumination differences in the image, affine transformations or large change in scale and rotation.

Furthermore, people have used neural networks [5], various classification algorithms [6] to train a model on a large dataset of images for logo recognition. These methods produce promising results; however, they are computationally expensive and time consuming (to build and train a model). There are simple algorithms which use edge detection [1] or morphological operations [7] for logo detection and recognition and have produced good results when we control some conditions in the input image.

Experiments:

The previous research work on VMR mainly focused on logo detection and recognition in frontal images of cars, so I collected a dataset of all frontal images of cars and decided to use the SIFT features based approach of make recognition presented in [3], which uses the descriptive SIFT features from the logo region of an image and compare it with pre-computed SIFT features of a logo template stored in a logo-dataset. I generated numerous keypoints in the logo region of two images using the SURF algorithm (Speeded-up Robust Features) implemented in Matlab which is an extension of SIFT algorithm, the match of features obtained is shown below:

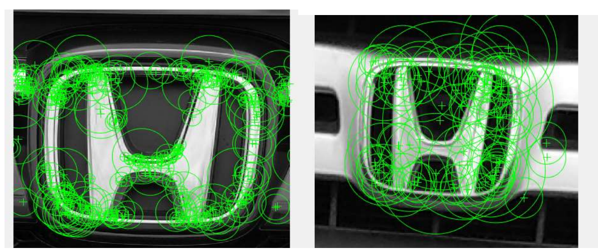

Fig. SURF features detected within the logo region of two query images

Fig. Matching SURF features between the two query images

Although it seems to match as several points, initially I was unable to obtain many matching keypoints in the two images, so I set the match method as 'Approximate' and a high value for 'MatchThreshold' parameter. If we look closely, we can see that the match information is not sufficient for accurate logo recognition. Below image shows a close view of results:

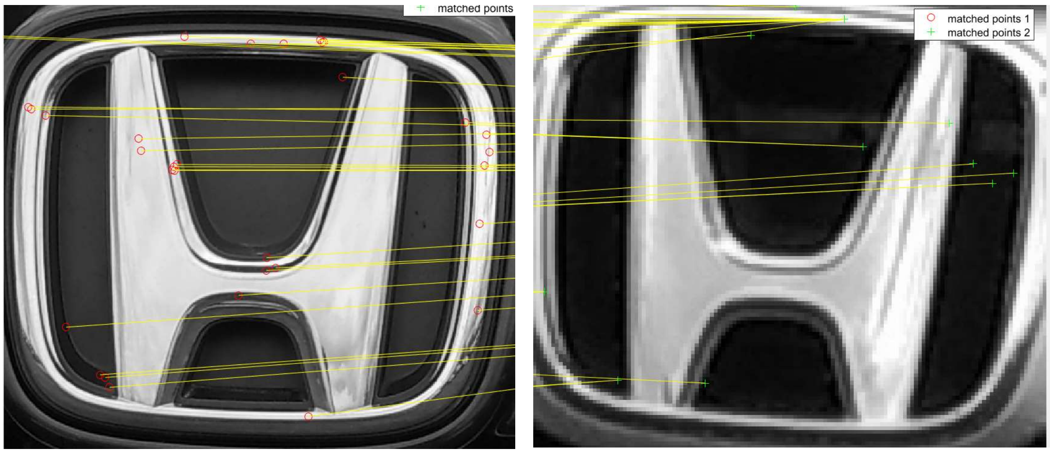

Fig. Close view of the matching SURF features in two images

We need atleast a close approximation of match to make a claim for the make of car. After testing this with several images I realized that features detected by SURF are not completely invariant to Scale and Rotation and are affected by change in varying scales.

Further, I tried to use the method described in [1], where the author uses edge detection and histogram projection to detect the location of logo. In this method, the author first determines the location of license plate and then looks for the logo in the region above the license plate using sobel filter for edge detection. The idea to detect the license plate was to find regions of text in the image. So, I used MSER features to detect MSER regions in the image and then performed some background subtraction to remove the non-text regions in image based on geometric properties and stroke width variation. I referred the matlab demo code "TextDetectionExample.m" for ocr detection.

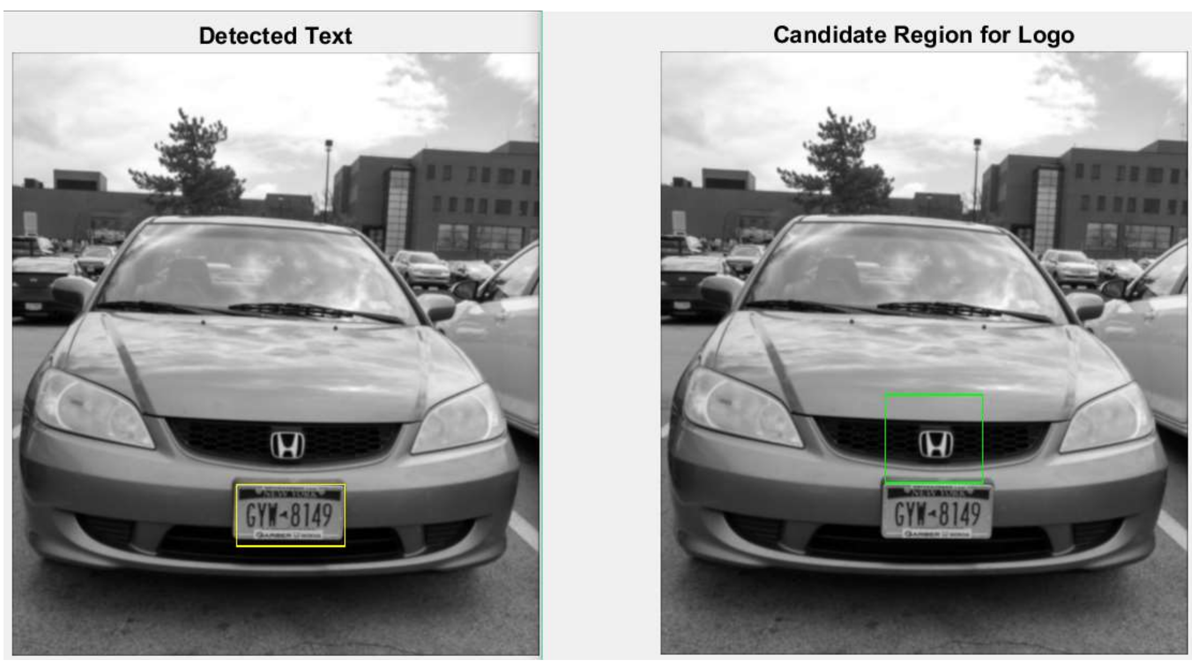

Fig. License Plate detection(left) and selecting a candidate region for logo(right) relative to the License Plate location

After selecting a candidate region, the author used some simple edge detection techniques using sobel edge detectors along with histogram of gradients to find a coarse location of logo. Although, I could not finish implementation of the algorithm mentioned in this paper, while working with the steps given by the author, I realized another way to overcome the problems faced by most of the methods I had been working on.

The idea was to use a template of logo (an image containing only logo) and using correlation to find the best match of this template in the query image. The steps followed in this approach are given below:

1. Select a logo template
2. Create a Scale space of this template by repetitively resizing and blurring the template with a Gaussian filter.
3. Resize the query image to size 500x500 and convert it to gray scale image and clean noise using Gaussian filter.
4. For each template in the scale space:
    4.1. Find the correlation between query_image and template_image.
    4.2. Find the peak of correlation.
    4.3. Crop the region around the peak point and store it in a temporary variable peak_region.
    4.4. Compute correlation again between the cropped peak_region and template.
    4.5. The correlation co-efficient obtained by above computation gives the degree of confidence.
5. Return the image with maximum degree of confidence.

Below are few images to explain the procedure:

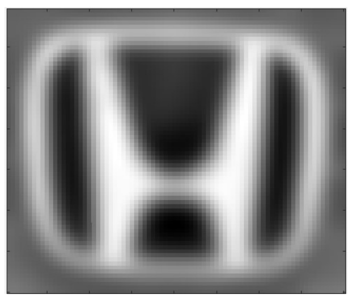

Fig. Step 1: Select a logo template

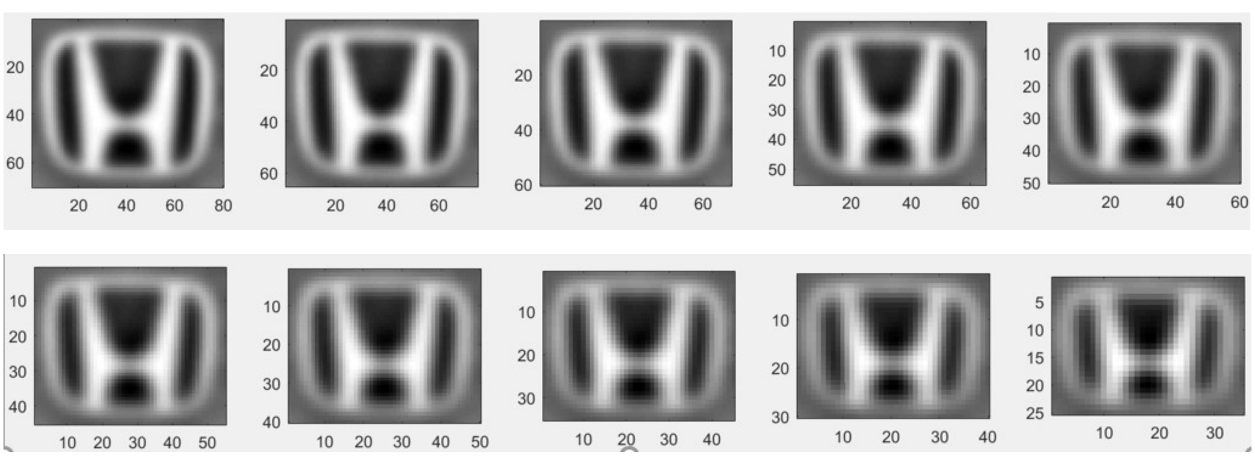

Fig. Step 2: Create a scale space from the logo template

Fig. Step 3: Query image converted to gray scale and resized to 500x500

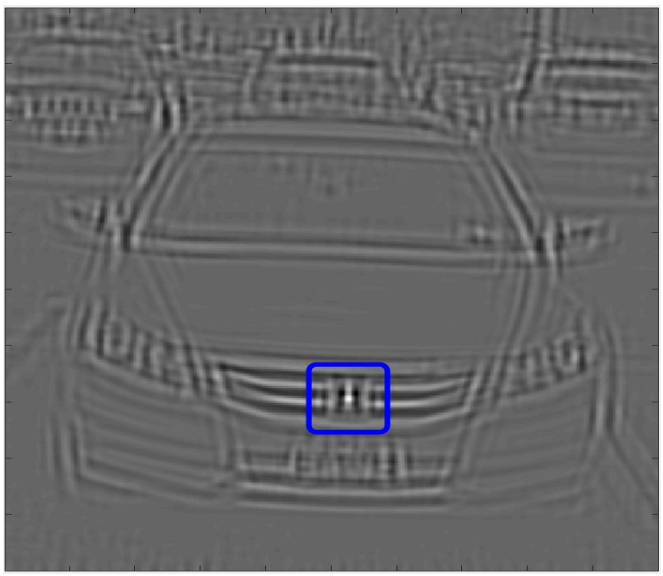

Fig. Step 4.1: Result of correlation with the closest matching template from scale space. The marked rectangle is the peak point of cross-correlation.

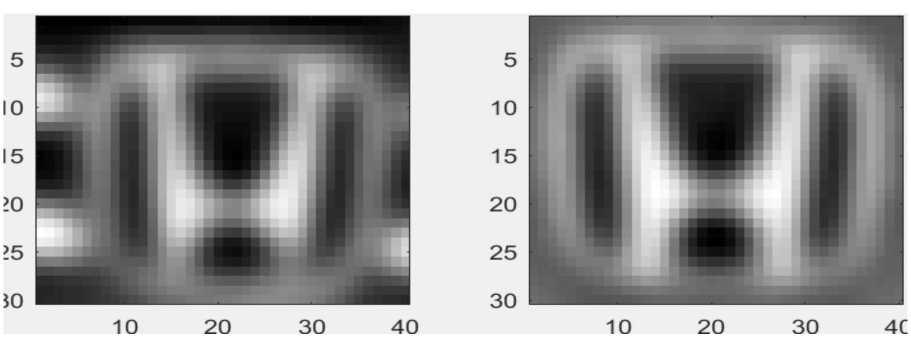

Fig. The cropped region around the correlation peak point detected in previous step(left) and a template from scale space with which it matched(right) matched to find the correlation coefficient (degree of confidence)

The degree of confidence obtained by the above match is 0.782, we repeat the same procedure with logo templates of other brands on the query image. The brand with the highest degree of confidence is the displayed with the name of manufacturer as shown below:

Fig: Output image with the detected logo and the recognized manufacturer

Results:

The algorithm given above worked for most of the images in the dataset I collected. Below is a table summarizing the results of implementation:

Brand	Total Images	Correctly Detected	Accuracy
Honda	122	112	91%
Toyota	119	105	88%

Out of the total 240 images, we had approximately 120 images of each brand and the logo detection algorithm worked correctly for almost all the images except a few (10 -15) images. We obtained correct detection for blur images, images with logos of different scales, logos at the front/back of a car, images with shadows on the logo region as shown below:

Fig. works good for logo in shadow(left) and logo of rear view of car(right)

The implementation did not work well with images in which the logo was completely under shadow or was not sufficiently illuminated. Below two images shows three such cases:

Fig. Logo detection fails when logo regions are hidden by shadow or are not properly illuminated.

The size of largest template we have is 70x80, those images with logo of size more than this are not detected

Fig. Query image with logo size more than the largest template.

Conclusion:

I learned several techniques we can use to detect any kind of object from natural images. An important step for any vision based application is cleaning noise and preprocessing the image as required by the application. In this project, I had to preprocess all the images in the dataset I collected to make sure that the images given algorithm have 1:1 aspect ratio and there is a car in the image. I also resize all the input images to size 500x500 and clean the background noise using a Gaussian filter. I faced several issues fixing the illumination changes in the images, these differences in illumination affected the results of correlation for most of the images. I fixed this issue by employing contrast adjustment methods for histogram equalization.

Bibliography:

[1] "A vehicle-logo location approach based on edge detection and projection" by Yang Liu; Shutao Li

[2] "A fast coarse-to-fine vehicle logo detection and recognition method" by Yunqiong Wang; Zhifang Liu; Fei Xiao

[3] "Vehicle Logo Recognition Using a SIFT-Based Enhanced Matching Scheme" by Apostolos P. Psyllos; Christos-Nikolaos E. Anagnostopoulos; Eleftherios Kayafas

[4] "Distinctive image features from scale-invariant keypoints" by David G. Lowe

[5] Vehicle logo detection using convolutional neural network and pyramid of histogram of oriented gradients by Wasin Thubsaeng; Aram Kawewong; Karn Patanukhom

[6] Vehicle logo recognition in traffic images using HOG features and SVM by D. F. Llorca; R. Arroyo; M. A. Sotelo