Vehicle Logo Recognition and Classification: Feature Descriptors vs. Shape Descriptors

Travis Burkhard (Tburk@stanford.edu
AJ Minich (Ajminich@stanford.edu)
Christopher Li (Chriswli@stanford.edu)

***Abstract*—We explore several image processing methods to automatically identify the make of a vehicle based focused on the manufacturer's iconic logo. Our findings reveal that large variations in brightness, vehicle features in the foreground, and specular reflections render the scale-invariant feature transform (SIFT) approach practically useless. Methods such as Fourier shape descriptors and inner structure mean square error analysis are able to achieve more reliable results.**

***Index Terms*—Vehicle logo recognition, SIFT, Fourier shape descriptors.**

I. INTRODUCTION

Vehicle manufacturer recognition offers a way to classify an automobile when license plate information is unavailable i.e. when the license plate is missing, covered, or forged. In this paper, we discuss and compare several methods of recognizing a manufacturer by way of the brand logo on the front or back of the car. We focus on methods of comparing the logo against a database of known brands, and using the calculated similarities to decide the most likely manufacturer. Although the traditional SIFT-based approach of using feature descriptors seems promising, as highlighted in [1] and [2] we show that the variations in light, size, and color of the logos significantly reduce the robustness of feature detection. We then discuss a novel technique using Fourier shape descriptors, which more naturally utilizes the varying shapes of the logos in determining the most likely manufacturers.

We ask that the reader note two assumptions we make about the test logo. First, we assume that the logo is seen head-on (no perspective change or rotation). Second, we assume that the logo is the largest object in the image. Thus, although we briefly discuss logo localization, these issues are largely outside of the scope of this paper.

II. IMAGE PREPROCESSING

In our approach, we use the fact that most manufacturers intentionally design brand logos to stand out on the car. That is, vehicle logos are usually brighter than the body of the vehicle, have well-defined edges, contain a closed shape (circular or rectangular) and that the most interesting portions of logo are around the edges. For more general scenarios, one approach in acquiring the logo image is to segment the logo via license plate detection [2]. We do not consider this constraint.

III. FEATURE DESCRIPTORS WITH SIFT

A natural approach to performing vehicle logo recognition under everyday conditions is to use SIFT [3]. Its invariance to perspective, shift, and rotation, in principle, allows us to match a given car logo image with a reference image with minimal pre-processing. This gives us one method to iteratively measure the similarity of a raw car logo image with various reference logo images in order to determine the most likely car manufacturer that corresponds to the logo. We can then use the distance ratio test between the keypoints of the raw image with the reference image to discard false positives. For elimination of outliers, the random sample consensus method (RANSAC) is typically employed.

The following Subsection describes our procedure with SIFT.

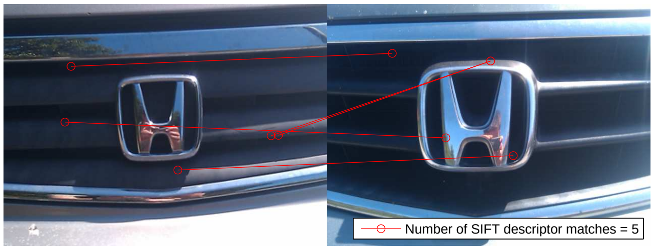
A. *Procedure with SIFT*

1) Convert the image to grayscale.
2) Apply a low pass filter followed by soft-coring.
3) Use a window to isolate the region of interest to the car logo.
4) Apply SIFT to the windowed image.
5) Apply ratio test.
6) Reduce outliers with RANSAC.
7) Compare the keypoint descriptors of the windowed image to those of a reference logo.
8) Measure the similarity between the raw logo image and various reference logo images. Decide on a certain manufacturer based on nearest neighbor (minimimum distance).

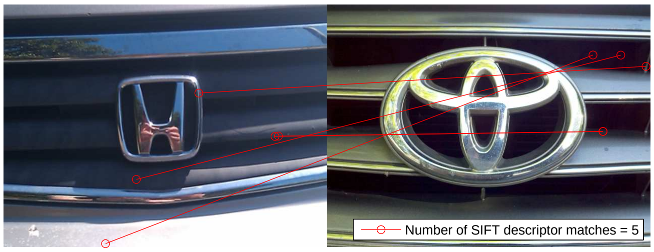
We found that the aforementioned procedure was largely insufficient for images captured under everyday conditions (i.e. outdoors with sunlight). Fig. 1a, Fig. 1b, and Fig. 1c show how conditions with large contrasts in light contribute to false positives between car logos, even with logos of the same manufacturer. We tested our approach using (1) logo graphics and (2) multiple close-up logo images acquired under various conditions (different lighting, different foreground features such as grill plates) as reference images. Neither approach delivered reliable results. We attribute this poor performance to the fact that chrome is a reflective surface, such that the gradient method behind SIFT doesn't work.

IV. SHAPE DESCRIPTORS WITH FOURIER TRANSFORMS

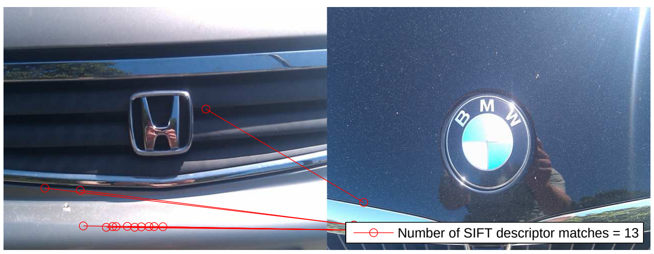
Fourier descriptors are a method of characterizing shapes based on their curvature.[4]. One key advantage of this method



(a) Illustration of poor SIFT performance for two images containing the same car manufacturer logo.



(b) SIFT false positives due to features in the foreground.



(c) SIFT mismatches under conditions with extreme glare.

over SIFT is that it is not as sensitive to distortion due to lighting effects from the environment, such as variations in lighting. The technique utilizes the shape of logos instead of individual descriptors. Fourier descriptors are translation and rotation-invariant. Scale invariance can be achieved via preprocessing and chip normalization [5]. The method has the special property that even if the logo isn't properly segmented, the comparison works as long as at least some salient curves on the outline are detected. It maps the contours of an image into discrete frequency components. We will refer to these components as Fourier descriptors. As with SIFT, we can determine the Euclidean distance between the Fourier descriptors of one image and those of another image to measure the similarity of two images.

We took several approaches with Fourier descriptors. In the first approach, we assume that features in the background have been sufficiently segmented.

A. Procedure with Fourier Descriptors 1: Convex Hull Approach

1) Low pass filter the image and apply soft coring.
2) Convert image to black and white and then apply Canny Edge detection.
3) Close the image using a disk-shaped structuring element.
4) Use region labeling to remove spurious components.
5) Find the convex hull of the region in the closed image with the largest convex area. This region is assumed to correspond with the logo.
6) Sample the convex hull. Each point (x_i, y_i) on the hull is converted to a sequence of complex-valued inputs $\{x_0 + jy_0, x_1 + jy_1, ..., x_{N-1} + jy_{N-1}\}$ into an N-point Fast Fourier Transform (FFT).
7) The N-point output of the FFT for image k, denoted as a vector $Y_k = [Y_0 + jY_1, Y_1 + jY_2, ..., Y_{N-1} + jY_{N-1}]$ is the Fourier descriptor itself.
8) Compare the Euclidean distance of Y_k with the Fourier descriptors of the M reference logo images in the database. The i^{th} manufacturer of the car is classified as $\arg\min_i \|Y_k - Y_i\|, i = 1, ...M$

B. Fourier Descriptor Experiment with Android and OpenCV

Image descriptors are well suited for mobile platforms due to their ability to compress large volumes amounts of data into a concise set of robust measurable qualities. This allows the mobile platform to either upload descriptors to a service or quickly iterate against a pre-cached database stored locally. We chose to implement the latter, compiling a database of Fourier descriptors offline and uploading them to the phone's SD storage.

Considering the convex hull results presented, and the fact that many logos have elliptical outlines, we chose to modify this procedure to determine the contours inside the logos. Segmentation is performed as in part IV-A, saving the normalized dilated edge map and the normalized convex area mask created in Step 5 above. An AND operation of the eroded convex outline and the inverted edge map isolates interior structures. Then edge detection is performed again to isolate the edges of the interior structures. These contours are then used as inputs to an FFT that calculates the Fourier descriptors.

The mobile platform has the ability to collect massive amounts of image data, including video files. However, performing full image detection on every frame brings frame rates to a halt, as even small images can take on the order of a seconds to segment, scale normalize, and produce a descriptor set. Therefore, we choose to do live segmentation on the phone, running at approximately one frame per second, depending on the hardware speed. Once the user has framed the target logo, he can then signal the rest of the processing pipeline, outlined in Fig. 2.

V. Shape Descriptors using Mean Square Error

Attempts to improve the accuracy rate of SIFT and Fourier descriptor based methods led us to develop a highly effective preprocessing method. Our methods provided accurate segmentation, thresholding, and scale normalization of the logo images. Rotation invariance is a non issue assuming that the logo is up right in the frame, and the camera has a gravity orientation sensor. The results of preprocessed images were so consistent that at a scale of $100px$ x $100px$ mean square error heuristic became computationally feasible for databases of moderate size.

For the MSE error analysis, we chose to reuse the interior negative space highlighting developed in the mobile Android/OpenCV pipeline. This allows us to reject the more common outer rim of most logos, and add decision weight to the unique interior logo structure. The process is very simple. A logo is segmented as before, however the convex mask is further eroded in order to ensure the outer most layer of detail in the logo (the external ring). Then the interior features are extracted as before, but no second edge detection is performed. This is illustrated in Fig. 1 . Instead the remaining area is used in a binary difference, and the minimum is difference is used to determine the best match.

Fig. 1: A thresholded Toyota reference logo (left) and its corresponding inner shapes (right), which will be used for mean-squared-error comparison.

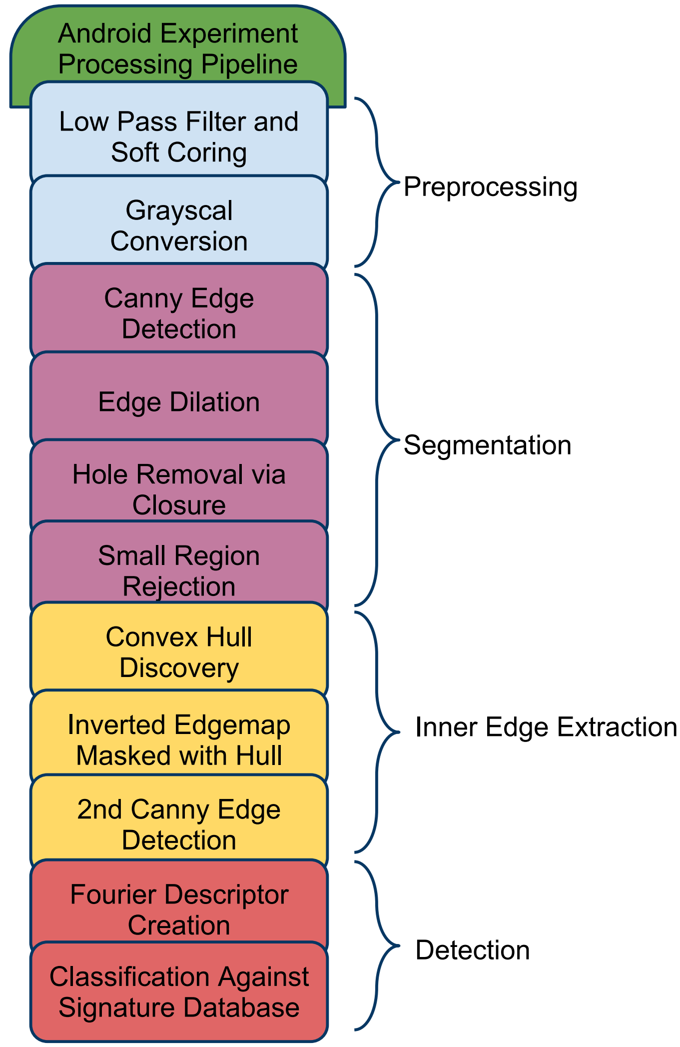


Fig. 2: An Android processing pipeline that implements Fourier shape descriptor comparison.

VI. Results

A. Fourier Descriptor Simulation Results

For images captured in more general settings, this method performs rather poorly. This approach only considers the outer contour of the car logo, and easily confuses car logos that have similar outer shapes (e.g. Toyota and BMW), as shown in Fig. 3. Without effective segmentation, this method becomes dependent on the logo being the dominant element in the image. As shown in Fig. 4, using Fourier descriptors on the convex hull alone is not a robust decision rule.

B. Android and OpenCV results

Although promising in theory, when supported by highly accurate segmentation routines, using Fourier descriptors to recognize inner structures on mobile phones is not accurate or robust enough to discriminate between the more complex inner structures of the logos. This area leaves room for future development. If a more robust feature set can be developed, this could prove promising. However as we see before, template matching proves to be a more robust method for logo recognition.

C. Mean Square Heuristic Results

Supplied with properly segmented and scaled images, a MSE decision heuristic preforms well as expected. In our exploration, MSE was the only method that gave us reliable results across the entire set of test images. As seen in Fig. 5, the MSE approach had a mean correct classification of 91.5%. However, a small percentage of results exhibit false classification, with regard to the logos with very similar layouts, such as Nissan, Toyota, and Honda.

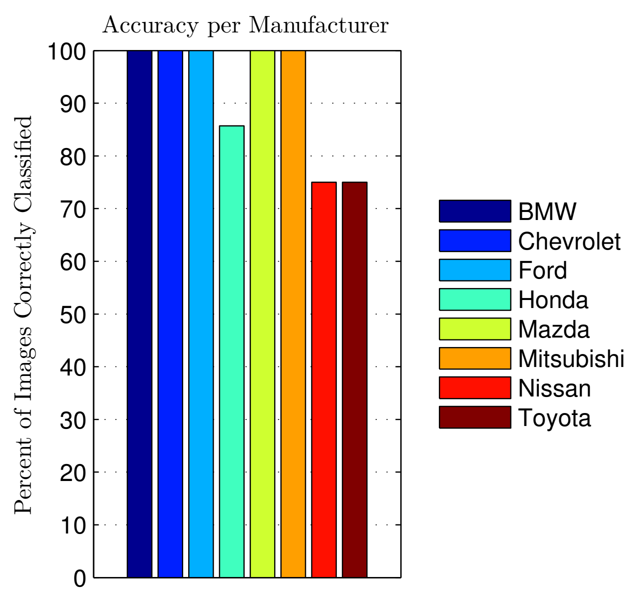


Fig. 5: Classification accuracy per car manufacturer using mean-squared-error comparison on interior shapes.

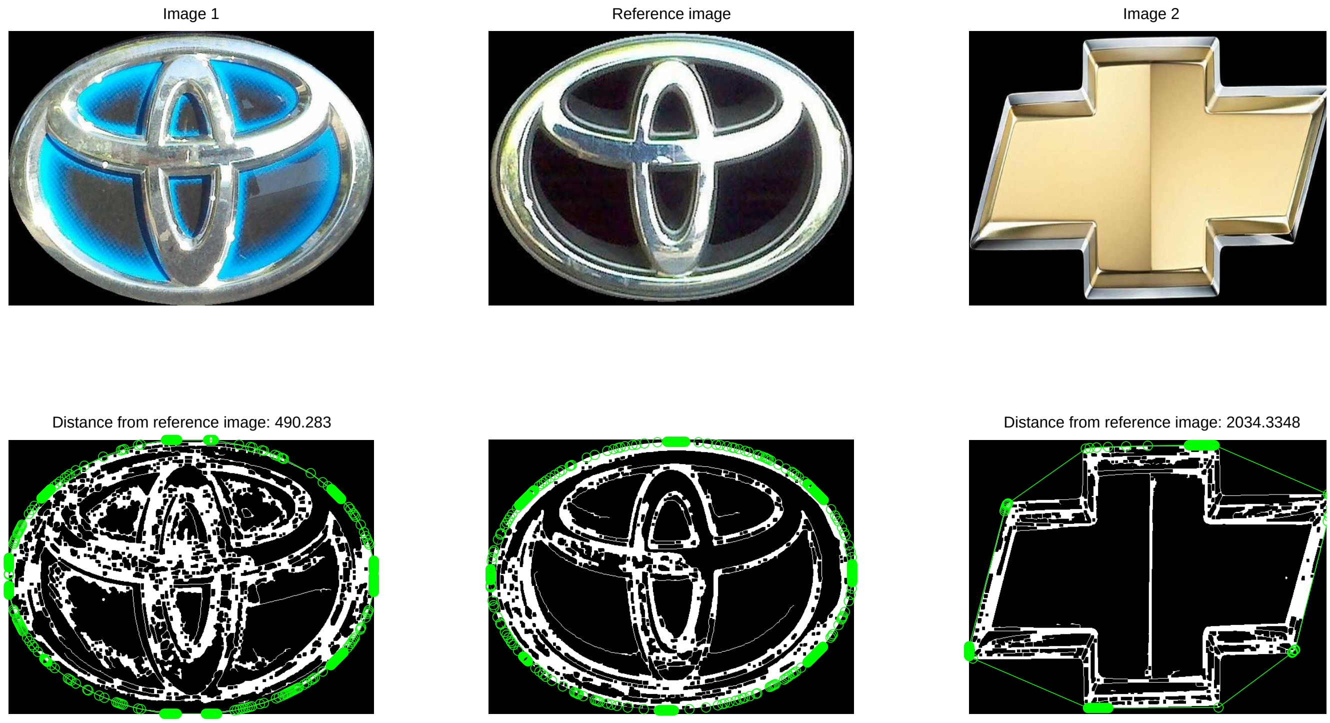


Fig. 3: For logos with similar outer shapes, acquiring Fourier descriptors from the convex hull performs reasonably.

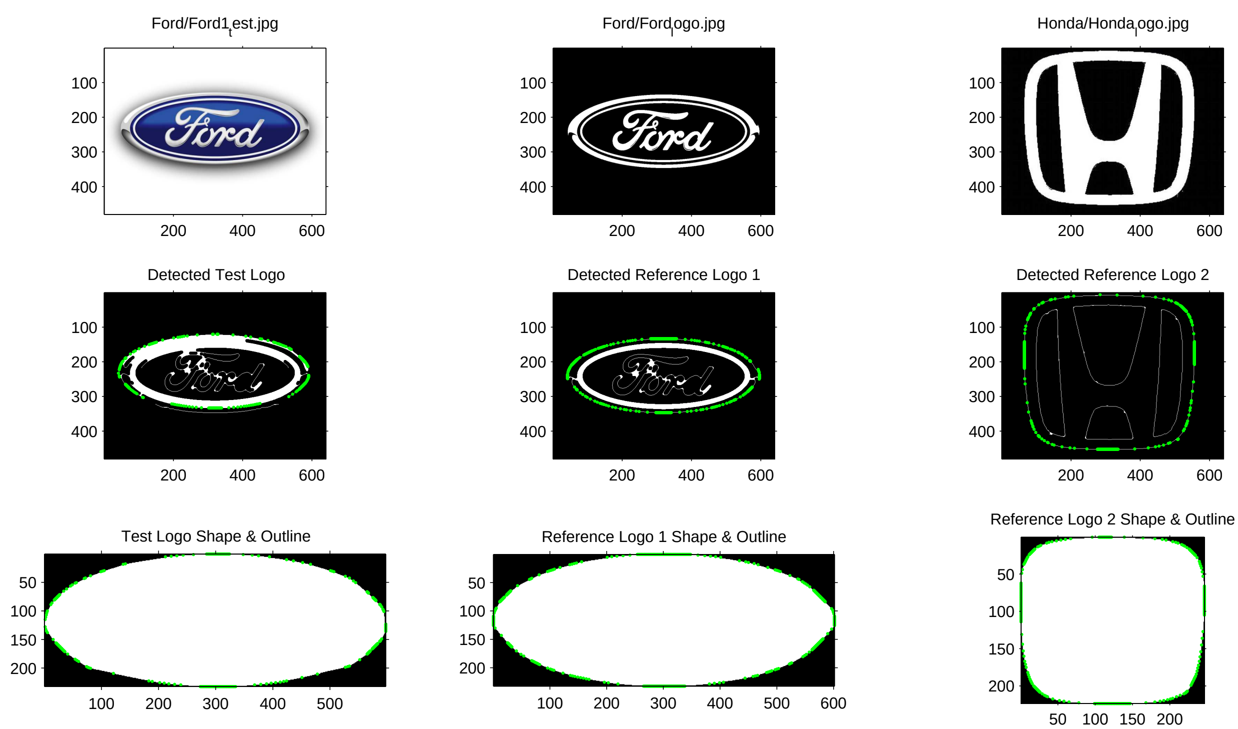


Fig. 4: Incorrect result from the Convex Hull Approach.

VII. Conclusions

A. Fourier Descriptors

Fourier descriptors seem to provide an interesting solution to this problem, bringing a non gradient based shift and rotation invariant solution. However, we found this novel way to classify shapes has a major fault. The Fourier descriptors proved to be to sensitive to irregularities and improper segmentation, especially when compared against much easier methods such as the MSE hueristic.

B. Interior Fourier Descriptors on Android

The mobile processing space still has yet to become developer friendly. The OpenCV port to Android provides a myriad of tools and customizability. However with the current state of JavaCV (an open source Java port of OpenCV stuck at version 1.1) and OpenCV currently reliant on SWIG wrappers for Java native code implementation, creating a development environment in this space is no where near trivial. Our experimentation shows that the high complexity algorithms of OpenCV can be run at acceptable frame-rates. However, our chosen experiment with internal structure Fourier descriptors proved to be nowhere near robust enough to merit a full analysis of its accuracy.

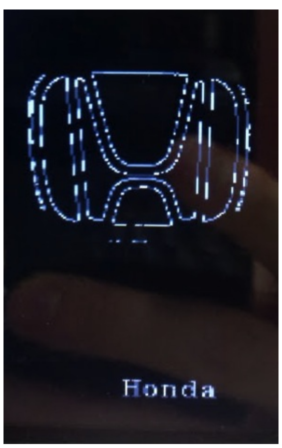


Fig. 6: A screenshot of a logo contour determined in realtime on an Android platform.

C. Mean Squared Error

The high accuracy of the mean-squared-error classification technique leads us to believe that the interior structures of the logos provide useful signatures for detecting similarity. In particular, although the segmentation algorithm discussed here does not guarantee accurate logo recognition, the interior shapes offer such unique data that they perform much higher than gradient-based features. We suggest that the analysis of these interior shapes, when combined with other classification methods, could provide even higher accuracy on larger sets of test images.

References

[1] A. P. . Psyllos, C. N. Anagnostopoulos, and E. Kayafas, "Vehicle Logo Recognition Using a SIFT-Based Enhanced Matching Scheme," *IEEE Transactions on Intelligent Transportation Systems*, 2010.

[2] A. Psyllos, C. N. Anagnostopoulos, and E. Kayafas, "Vehicle model recognition from frontal view image measurements," *Computer Standards and Interfaces*, 2010.

[3] D. Lowe, "Object recognition from local scale-invariant features," in *International Conference on Computer Vision 2*, 1999, pp. 1150–1157.

[4] D. Zhang and G. Lu, "A Comparative Study on Shape Retrieval Using Fourier Descriptors with Different Shape Signatures," *Gippsland School of Computing and Information Technology*.

[5] B. I. of Technology, "Image analysis exercise 1:fourier descriptors," 2009.

[6] H. Bay, T. Tuytelaars, and L. Gool, "Surf: Speeded up robust features," in *European Conference on Computer Vision*, no. 9, May 2006.

[7] T. Kato, Y. Ninomiya, and I. Masaki, "Preceding vehicle recognition based on learning from sample images," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, pp. 252–260, Dec 2002.

Implementation Activity	**Travis Burkhard**	**Christopher Li**	**AJ Minich**
MATLAB			
classify_images.m			√
classify_manager.m			√
crop_binary.m			√
get_fft_distance.m		√	
get_logo_features.m	√		√
get_outline_and_inner_structure.m	√		√
im_calc_mse.m			√
preprocess.m		√	√
run_comparison_test.m		√	
run_single_test.m		√	
segment_logo.m			√
Android			
Compiling	√		
OpenCV Development	√		√
OpenCV Integration	√		

Fig. 7: Distribution of work with respect to implementation.