Vehicle Make & Model Recognition

by
Zhihao DAI

COP507 Computer Vision & Embedded Systems
Coursework Report

Loughborough University

Jan. 2020

Abstract

In this coursework, I implement a JPEG Image Compression Simulation using MATLAB as frontend GUI and Python as backend JPEG CODEC. There are 2 simulation parameters K and Q' in the application. Several specific design considerations are introduced to the implementation, inclding an end-to-end MATLAB interface, an "Video Compression" functionality and DCT as Matrix Computation. I conclude that both K and Q' can significantly affect the quality of the compressed image.

Contents

List of Figures

List of Tables

List of Listings

Chapter 1

Introduction

Automatic Number Plate Recognition (ANPR) systems are widely used for policing, traffic monitoring and access control. They have proven to be accurate and efficient under most scenarios. However, ANPR systems are vulnerable to plate cloning, forgery or erosion.

A Vehicle Make & Model Recognition (VMMR) system receives an image of a vehicle as input and outputs the make and model of that vehicle. Such system could strengthen the security of existing ANPR systems by providing a matching between vehicle types and number plates. For example, in access control, if the number plate is not registered under the detected vehicle type, a security warning is raised and manual intervention is required.

In this paper, we design and implement a VMMR system. The input to the system is a cropped frontal image of a vehicle and the output is the make and model of the vehicle.

1.1 Related Work

Due to the significance of VMMR systems, many approaches have been proposed for building VMMR systems in recent years. Petrovic and Cootes [11] extracted simple features such as Sobel Edge Response, Edge Orientation, Square Mapped Gradients from images in the database. Features are then represented and stored either in full dimension or in low dimension through Principal Component Analysis (PCA). Given a new image, the VMMR system predicts the vehicle type by finding the closest match in dot product distance. Their experiments on a dataset of 1132 frontal images of 77 vehicle classes showed that direct matching by Square Mapped Gradients features achieved the lowest vertification error of 3.5%.

AbdelMaseeh et al. [1] observed that unlike most object recognition tasks, VMMR poses a challenge of distingushing between similar classes under the same category (ie. vehicle).

Based on this observation, they proposed the combination of global and local descriptors for VMMR. While global shape descriptors capture differences across categories, local shape and appearance descriptors for segmented regions capture inter-class varieties. An image is matched to the class with the smallest weighted sum of global and local dissimilarity measures.

Pearce and Pears [9] suggested using Harris corner detectors [4] for features extraction and either K-Nearest Neighbour (KNN) or Naive Bayes Classifier for classification in VMMR systems. Local Harris strengths are computed through recursively dividing the image into quadrants and computing the sum of Harris corner response for each quadrant. Such features are then normalised through being devided by the sum of higher level strengths. For an input image of 150 by 150, a feature vector of Locally Normalised Harris Strengths (LNHS) of depth 5 is retrived and only one-twentieth the size of the original Harris corner response. Their experiments on a dataset of 262 frontal images of 74 vehicle classes showed that LNHS with Naive Bayes Classifier achieved the highest accuracy of 96%. Using LNHS as features speeds up the training of a classifier and does not reduce the accuracy.

Siddiqui et al. [12] proposed using Speeded Up Robust Features (SURF) [2] for features extraction and Support Vector Machines (SVM) for classification. Following Sivic and Zisserman's work on Bag-of-Features method [13], a dictionary (bag) of SURF features was constructed using K-Means clustering algorithm. An image can be then transformed into a fixed-length vector of visual words occurances and be fed into a SVM classifier for vehicle type recognition. High accuracy score of 94.84% was obtained on a large dataset of 6601 frontal images of 29 vehicle classes.

Zafar et al. [16] observed that dimensionality reduction methods used in many VMMR systems such as Principal Component Analysis (PCA) enhances the inner-class variance and can lead to miss-classification. In their setting, the raw pixel values of the image is directly projected to low-dimension space through Two Dimensional Linear Discriminant Analysis (2D-LDA) [6]. A match is found by minmizing the Euclidean distance to those in the training images set. The usage of 2D-LDA instead of PCA solves the variance problem by maximizing the ratio of intra-class variance to the inter-class variance. An accuracy score of 91% was obtained on a dataset of 271 frontal images of 25 vehicle classes (8 images per class for training and the rest for validation).

Zafar et al. [15] later proposed using localized Contourlet transform for features extraction, 2D-LDA for dimensionality reduction, and SVM for classification. They reported a boosted accuracy of 96% on the same frontal car images dataset in [16].

Fraz et al. [3] introduced an innovative framework of Mid-Level-Representation of densely sampled features into VMMR. The framework starts by extracting patches around key-points detected by Difference of Gaussians (DoG) detector. For each extracted patch, A set of

Scale-Invariant Feature Transform (SIFT) [7] feature descriptors are computed and reduced dimensionality by PCA. Fisher Vector [5], a Mid-Level-Representation (MLR), for the patch is then generated based on Gaussian Mixture Model (GMM), following Perronnin et al.'s work [10]. Fisher Vector for patches in images within the same class are visual words and collectively form a sub-lexicon. A lexicon of the training set images is essentially a collection sub-lexicons of all classes. Given a new image, the VMMR system extracts patches from the image, assigns each patch to a visual word by Euclidean distance within each sub-lexicon, classifies the image to the class (sub-lexicon) with the highest sum of similarity score of the word-patch matches. Fraz et al. reported an accuracy of 97.60% on the dataset used in [15] and 84.31% on a new dataset. The new dataset, coined 'Loughborough Cars (LC) Dataset', is composed of 1537 frontal images of 75 vehicle classes.

1.2 Dataset

There is a diverse set of datasets for VMMR task and Tafazzoli et al. [14] presented a thorough survey of them. Our proposed system is trained and evaluated on a superset of the dataset in [16, 15, 3] of 530 frontal images from 27 vehicle make and model classes. For each class, 6 images are used for training and the rest for validation. Both the training set and validation set are pre-processed to extract Regions of Interest (ROI). See Section 3.1 for more details.

1.3 Comparison to Our Method

In this paper, we make use of Raw Image Pixels, Sobel Edge Response and Square Mapped Gradients following Petrovic and Cootes's [11] work, Locally Normalised Harris Strengths (LNHS) from Pearce and Pears's work [9], and Bag of Speeded Up Robust Features (SURF) from Siddiqui et al.'s work [12] interchangeably in our features extraction module. We use Principal Component Analysis (PCA) for optional dimensionality reduction module and either K-Nearest Neighbour (KNN) or Support Vector Machine (SVM) for classification module.

Despite having a smaller number of 4 training images per vehicle class compared to 8 in Zafar et al.'s work [16] and 10 in Zafar et al.'s work [15] and simplicity of features computation compared to Mid-Level-Representation in Fraz et al.'s work [3], our method achieves a higher accuracy score of 98% on the validation set.

Figure 1.1: Samples of All 27 Vehicle Make and Model Classes in the Dataset.

Chapter 2

System Design

2.1 Assumptions

Several assumptions are made in our design and implementation of VMMR system.

1. The input to the system are frontal images of vehicles.
2. Region of Interest (ROI) can be extracted from the input image based on a pre-labeled bounding box of number plate.
3. The true make and model label for any input image is one of the 27 classes outlined in Section 1.2.

2.2 Block Diagram

A block diagram of our proposed VMMR system is presented in Figure 2.1.

At the first stage, **Features Extraction Module** extracts features from the input image I into a fix-length vector F. **Dimensionality Reduction Module** is optional and reduces the dimensionality of the high-dimensional vector F into low-dimensional vector F'. **Classification Module** is essentially a multi-class classifier that learns to assign incoming feature vectors to their corresponding true labels.

Figure 2.1: Block Diagram of Our Proposed VMMR System.

2.3 Features Extraction

A variety of features are interchangeably computed in Features Extraction Module. Among them, Raw Image, Sobel Edge Response (SER), Square Mapped Gradients (SMG) are parameter-free and first proposed to be used for VMMR by Petrovic and Cootes in [11]. Locally Normalised Harris Strengths (LNHS) is first proposed by Pearce and Pears in [9]. Bag of Speeded Up Robust Features (BSURF) is first used by Siddiqui et al. for VMMR in [12].

In Section 3.4, the performance of the above features are compared and the effects of parameters of LNHS and BSURF on performance are examined.

2.3.1 Raw Image

Raw Image features are the image pixel values themselves. Hence, $F = I$.

2.3.2 Sobel Edge Response (SER)

SER (also named Sobel Gradient Map) is a map of weighted sum of pixels at 3-by-3 neighborhood.

$$S_{i,j} = \sum_{p=1}^{3}\sum_{q=1}^{3} W_{p,q} I_{i+p-2,j+q-2} \tag{2.1}$$

where in y-direction, $W = W^y$ is specified as follows.

$$W_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{2.2}$$

In x-direction, $W = W^x = W^y$.

The final feature vector F is the concatanation of S^x and S^y.

$$F = (S^x, S^y) \tag{2.3}$$

2.3.3 Square Mapped Gradients (SMG)

SMG describes the parallel and diagonal components of change in Sobel Edge Response. The parallel component M^p and diagonal componet M^d are computed as follows.

$$M^p_{i,j} = \begin{cases} 0, & \text{if } S^x_{i,j} = 0 \text{ and } S^y_{i,j} = 0 \\ \frac{{S^x_{i,j}}^2 - {S^y_{i,j}}^2}{{S^x_{i,j}}^2 + {S^y_{i,j}}^2}, & \text{otherwise} \end{cases} \tag{2.4}$$

$$M_{i,j}^d = \begin{cases} 0, & \text{if } S_{i,j}^x = 0 \text{ and } S_{i,j}^y = 0 \\ \frac{2 \cdot S_{i,j}^x S_{i,j}^y}{{S_{i,j}^x}^2 + {S_{i,j}^y}^2}, & \text{otherwise} \end{cases} \tag{2.5}$$

The final feature vector F is the concatanation of M^p and M^d.

$$F = (M^p, M^d) \tag{2.6}$$

2.3.4 Locally Normalised Harris Strengths (LNHS)

LNHS is a recurisve structure of Harris corner [4] features representaiton. Given an image, Harris corner strengths $M = \{M_c\}$ are first computed following Noble's suggsted formulation [8].

$$M_c = \frac{I_x^2 I_y^2 - (I_x I_y)^2}{I_x^2 + I_y^2} \tag{2.7}$$

where I_x and I_y are smoothed image derivatives in x-direction and y-direction respectively.

Local Harris corner strengths L are computed through recursively dividing the M into quadrants and computing the sum of Harris corner strengths M_c for each quadrant. Local strengths are then normalised into a vector of LNHS through being devided by the sum of higher level strengths.

For example, for depth of 1, M is first divided into 4 sub-matrices $M_1, M_2, \ldots, M_4$.

$$M = \begin{bmatrix} M_1 & M_2 \\ M_3 & M_4 \end{bmatrix} \tag{2.8}$$

The overall LNHS feature vector F is equal to LNHS vector of depth 1, L_1.

$$L_1 = \{\frac{sum(M_i)}{\sum_i sum(M_i)} | i \in \{1, 2, 3, 4\}\} \tag{2.9}$$

For depth of 2, the above 4 sub-matrices $M_1, M_2, \ldots, M_4$ are further divided into quadrants respectively.

$$M_1 = \begin{bmatrix} M_{1,1} & M_{1,2} \\ M_{1,3} & M_{1,4} \end{bmatrix} \tag{2.10}$$

LNHS vector of depth 2 are computed through dividing the sum of each quadrant by the sum of its higher level quadrant.

$$L_2 = \{\frac{sum(M_{i,j})}{\sum_j sum(M_{i,j})} | i, j \in \{1, 2, 3, 4\}\} \tag{2.11}$$

The overall LNHS feature vector F is concantaion of LNHS vector of depth 1 and 2.

$$F = (L_1, L_2) \tag{2.12}$$

Depth is the sole parameter in extracting LNHS features.

2.3.5 Bag of Speeded Up Robust Features (BSURF)

To compute a BSURF vector, SURF features [2] are first detected and extracted from the image. A bag of visual words can be constructed by grouping all SURF feature descriptors from the training set into T clusters using K-Means algorithm.

Given any image I, each SURF descriptor extracted can then be assigned to the nearest among the above T clusters. The BSURF feature vector F for I is a vector for visual words occurrences of fixed length T.

The number of clusters T is the most important paramter in BSURF and is examined in our experiments.

2.4 Dimensionality Reduction

Features extracted from the images are usually correlated and can be reduced in dimensionality. Such reduction speeds up the training and reduces the complexity of a classifier, which help prevents over-fitting issues.

2.4.1 Principal Component Analysis (PCA)

PCA maps high-dimensional data into a new low-dimensional coordinate system through Singular Value Decomposition (SVD).

Durining training, a high-dimensional matrix X is formed, where each row is a feature vector F extracted from a image in the training set. SVM docomposes X into a product of 3 matrices.

$$X = U\Sigma W^T \tag{2.13}$$

where U is a m-by-m matrix, Σ is a m-by-n diagonal matrix and W^T is the transpose of W, a n-by-n matrix.

X is then reduced in dimensionality to produce X'.

$$X' = XW_L \tag{2.14}$$

where W_L only preserves the first L columns of W.

Likewise, for any 1-by-n feature vector F, a new vector F' reduced in dimensionality can be derived.

$$F' = FW_L \tag{2.15}$$

2.5 Classification

2.5.1 K-Nearest Neighbour (KNN)

2.5.2 Support Vector Machine (SVM)

Chapter 3

Experiments and Results

Environment, etc.

3.1 Pre-processing

3.2 Cross-Validation

3.3 Merits of Performance

3.4 Effects of Features Extraction Methods

3.5 Effects of Dimensionality Reduction Methods

3.6 Effects of Classification Methods

Chapter 4

Convolution Neural Network Model

4.1 Architecture

4.2 Overfitting Issues

4.3 Data Augmentation

Chapter 5

Discussion

5.1 Conclusions

5.2 Future Work

References

[1] Meena AbdelMaseeh, Islam Badreldin, Mohamed F Abdelkader, and Motaz El Saban. Car Make and Model Recognition Combining Global and Local Cues. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pages 910–913. IEEE, 2012.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded Up Robust Features. In *European Conference on Computer Vision (ECCV)*, pages 404–417. Springer, 2006.

[3] Muhammad Fraz, Eran A Edirisinghe, and M Saquib Sarfraz. Mid-Level-Representation based Lexicon for Vehicle Make and Model Recognition. In *2014 22nd International Conference on Pattern Recognition (ICPR)*, pages 393–398. IEEE, 2014.

[4] Christopher G Harris, Mike Stephens, et al. A Combined Corner and Edge Detector. In *Alvey Vision Conference*, volume 15, pages 10–5244. Citeseer, 1988.

[5] Tommi Jaakkola and David Haussler. Exploiting Generative Models in Discriminative Classifiers. In *Advances in Neural Information Processing Systems (NIPS)*, pages 487–493, 1999.

[6] Ming Li and Baozong Yuan. 2D-LDA: A Statistical Linear Discriminant Analysis for Image Matrix. *Pattern Recognition Letters*, 26(5):527–532, 2005.

[7] David G Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[8] Julia Alison Noble. *Descriptions of Image Surfaces.* PhD thesis, University of Oxford, 1989.

[9] Greg Pearce and Nick Pears. Automatic Make and Model Recognition from Frontal Images of Cars. In *2011 8th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 373–378. IEEE, 2011.

[10] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the Fisher Kernel for Large-Scale Image Classification. In *European Conference on Computer Vision (ECCV)*, pages 143–156. Springer, 2010.

[11] Vladimir S Petrovic and Timothy F Cootes. Analysis of Features for Rigid Structure Vehicle Type Recognition. In *British Machine Vision Conference (BMVC)*, volume 2, pages 587–596, 2004.

[12] Abdul Jabbar Siddiqui, Abdelhamid Mammeri, and Azzedine Boukerche. Real-Time Vehicle Make and Model Recognition Based on a Bag of SURF Features. *IEEE Transactions on Intelligent Transportation Systems*, 17(11):3205–3219, 2016.

[13] Josef Sivic and Andrew Zisserman. Video Google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision (ICCV)*, page 1470. IEEE, 2003.

[14] Faezeh Tafazzoli, Hichem Frigui, and Keishin Nishiyama. A Large and Diverse Dataset for Improved Vehicle Make and Model Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1–8, 2017.

[15] Iffat Zafar, Eran A Edirisinghe, and B Serpil Acar. Localized Contourlet Features in Vehicle Make and Model Recognition. In *Image Processing: Machine Vision Applications II*, volume 7251, page 725105. International Society for Optics and Photonics, 2009.

[16] Iffat Zafar, Eran A Edirisinghe, S Acar, and Helmut E Bez. Two Dimensional Statistical Linear Discriminant Analysis for Real-Time Robust Vehicle Type Recognition. In *Real-Time Image Processing 2007*, volume 6496, page 649602. International Society for Optics and Photonics, 2007.

Appendix A

Source Code