

LOUGHBOROUGH UNIVERSITY

COP507

COMPUTER VISION & EMBEDDED SYSTEMS

---

# Report on Vehicle Make and Model Recognition System

---

*Author:* EKOW MENSAH  
(B717426)



**Loughborough  
University**

January 8, 2018

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Preprocessing Tasks</b>	<b>3</b>
2.1	CCTV camera located at the side of a motor way . . . . .	3
2.1.1	Skewing & Rotation . . . . .	3
2.1.2	Brightness & Colour Correction . . . . .	4
2.1.3	Motion blur removal . . . . .	5
2.1.4	Shake Removal . . . . .	6
2.2	Dedicated camera installed at entrance to indoor car park . .	8
2.2.1	Noise Removal . . . . .	8
2.2.2	Improving Low resolution images . . . . .	9
2.2.3	Removal of Barrelling Effect . . . . .	10
2.2.4	Contrast Stretching . . . . .	11
<b>3</b>	<b>The proposed VMMR system</b>	<b>12</b>
3.1	High level block diagram . . . . .	12
3.1.1	Loaded images . . . . .	12
3.1.2	Feature Detection . . . . .	12
3.1.3	Feature Extraction . . . . .	13
3.1.4	Creation of Bag of Features . . . . .	13
3.1.5	Training . . . . .	13
3.1.6	Prediction of classes (Testing) . . . . .	13
3.1.7	Results . . . . .	13
3.2	System Design . . . . .	14
3.2.1	Loading the images . . . . .	14
3.2.2	Feature detection algorithm . . . . .	14
3.2.3	Extraction of SURF features . . . . .	14
3.2.4	Creation of Bag of Features . . . . .	15
<b>4</b>	<b>Experiments and Results</b>	<b>15</b>
<b>5</b>	<b>Reference List</b>	<b>16</b>

# 1 Introduction

Make and Model Recognition (VMMR) has been a subject area of interest within computer vision both in industry and academia. The importance of the ability to recognise a variety of vehicles spans across different disciplines and applications such as driver assistance, traffic monitoring, law enforcement and surveillance. As stated by Dehghan et al (2017), this categorisation task has been a problem for classical computing however, a result of great advances in artificial intelligence, this task has been made possible with accuracy rates above 90 percent. The task given was to design and implement a VMMR system with the capability to recognise the make and model of a vehicle based on only the front view (that is, headlights, grill and bumper). Based on the image of a vehicle fed as input, the output of the system should be the make and model of the vehicle. This report explains a practical use of the VMMR system that was implemented, furthermore, it explains the design of the system and any performance restrictions or performance enhancements that will be expected should the system implemented in hardware.

There some basic processes that are essential for any computer vision application. These include pre-processing the acquired images, feature detection and feature extraction. Pre-processing involves making the image data conducive for feature extraction. Some of these processes includes noise reduction, morphological operations, colour correction and image re-sampling. Feature extraction is the process by which key points are obtained from the image. These key points consist of blobs, edges and corners.

## 2 Preprocessing Tasks

This section discusses the pre-processing tasks that are required to implement the Vehicle Make and Model System in practical scenarios.

### 2.1 CCTV camera located at the side of a motor way

As a consequence of this system being implemented outdoors, there are certain factors that cannot be controlled prior to the CCTV footage being taken. These factors include illumination, motion blur, problems with focus and the camera being placed at an angle which affects processing of the image. The following subsections explain solutions to these problems found when processing the images taken by the camera.

#### 2.1.1 Skewing & Rotation

Considering the camera is at the side of the motor way, one thing to consider is that the angle at which the camera is placed. Therefore, before performing any feature detection or extraction, the images taken from that camera must be set to the right orientation. This could be done by skewing the image or rotating the image as this will set the image to the right orientation. Traditional mathematical algorithms and formulas exist which can be used to correct the angle of rotation or the degree of skewing required. Considering that since an image can be represented as a matrix of pixel values, affine transformation matrices could be applied to the image to correct the problem (Gonzales & Woods, 2008).

Before these algorithms can be applied to the images, it must be detected whether there is a need to apply skew correction to the images captured. Patel et al. (2015) discuss an approach for skew angle detection and correction

based on the radon transform. The radon transform was used because of its accuracy and low computational cost. The radon function calculates projections of the image matrix in particular directions (Patel et al, 2015). For an image matrix  $f(x,y)$  a projection is considered as a collection of line integrals. The line integrals are calculated from various sources along beams, lateral paths, in a specific direction. The skew detection algorithm proposed by Patel et al (2015) is as follows. First of all, the input image is binarised and morphological filling operations are applied. After that, edges are detected using the canny edge detector. This process is used for locating relevant transitions within the image. The radon transform is then applied on the output of the edges detected. Next, a maximum value is searched for within the radon transform coefficients. After that, a skew angle is found from the maximum intensity value that was found from the radon coefficients. The image is then rotated with the skew angle.

### 2.1.2 Brightness & Colour Correction

Another factor to consider is the lighting conditions. The reason being that, in the day time, pictures taken may or may not have the appropriate lighting. In the night time, the images taken may be very dark. Therefore, in order to make the images conducive for training, the images will need to be made brighter. A remedy to this problem could be applying a brightness adjustment algorithm. According to Szeliski (2010), this can be done by multiplying or adding the image by the constant.

$$g(x) = \alpha f(x) + \beta$$

Where  $\alpha$  and  $\beta$  are parameters which influence the contrast and brightness of the image and where  $\alpha$  is greater than zero.

Alternatively, colour balancing and correction algorithms can be used to improve the illumination and contrast of the image. Weng et al (2004) introduce

a novel approach of improving white balance in still images. This algorithm is needed especially for images taken at night because, depending on the colour temperature of the street light. The image will appear reddish if the colour temperature of the light source is low, conversely, the image will appear blue if the colour temperature of the light source is high (Weng et al, 2014). Therefore, improving the white balance will make the image appear as though it was captured under canonical lighting.

The proposed solution works in two stages that is, white point detection and white point adjustment. utilises a dynamic threshold for identifying white points within the image. The process begins by converting the image from the  $RGB$  colour space to the  $YC_bC_r$ . Depending on the colour characteristics, a near white region composed of several candidate reference white points is chosen. The candidate points are chosen by computing the mean values of  $C_b$  and  $C_r$  (Weng et al, 2014). The white reference points are then selected depending on the brightness values of candidate white points (top 10 percent) near the white region. After the white reference points have been chosen, the Von Kries model is used to fine-tune the image. This model is used to scale each individual  $R$ ,  $G$  and  $B$  channel separately. Channel gains are obtained from the average values of reference white points. Luminance is maintained throughout the image at the same intensity by using the maximum luminance value to compute channel gains. Each pixel value is adjusted by multiplying each pixel with the computed gain (Weng et al, 2014).

### 2.1.3 Motion blur removal

Motion blur is a phenomenon which occurs when there are inconsistencies between the speed of the moving object and the rate at which the camera captures frames. The problem with motion blur is that, it attenuates the image signal. Removing the motion blur from the image involves two processes. Firstly, estimating the amount by which the image has been blurred and thereafter, recovering a more realistic image using deconvolution (point-

wise division in the frequency domain). Cho (2010) proposes a method of deconvolution referred to as iterative distributed reweighting (IDR) for removing motion blur. This method enhances the visual quality of reconstructed images as compared with traditional deconvolution algorithms used for removing motion blur.

Cho (2010) suggests the estimation of the blur kernel by examining edges within the blurred image. The edges within the blurred image encode estimations of the blur kernel from which the blur can be removed using the inverse radon transform. Using the edge information in this case will yield desirable results because, the object of interest is a man made object and man made objects consist of many edges. As a result of the loss of high frequency information, prior information concerning natural images is used to fill out the missing information. This is done using content aware image priors rather than sparse priors because, content aware prior adjusts its features to local textures and hence, it increases the quality of textures within the reconstructed images (Cho, 2010). After this has been completed, a maximum a posteriori (MAP) estimator is used to restore the image using piecewise smooth features, however, since the MAP does not always accurately reconstruct rich textures, IDR is used to improve the depiction of rich textures.

#### **2.1.4 Shake Removal**

In the process of capturing images, the camera could be susceptible to shaking especially in windy conditions. This is a problem because, the camera shaking causes non-uniform blurring cannot be handled effectively by blind convolution algorithms (Hirsch et al, 2011). They suggest a fast single image blind deconvolution algorithm which is a combination of both constraints of Projective Motion Path Blur (PMPB) models and the effectiveness of the Efficient Filter Flow (EFF) framework. In order to recover a sharper version of the image, EFF estimates a non-static blur as the total of various blur

patches. The blur kernels are created using homographies. The homographies are then applied to a grid of pixels. This makes it possible to generate possible camera shakes via a combination of various homographies of a point grid (Hirsch et al, 2011). Different views of a point grid are created by using a homography. This splits the views into local blur kernels which are constrained to only linear combinations.

Given a blurred image, an unknown sharp image can be reconstructed in two steps: (1) blur estimation for non-stationary PSFs and (2) sharp image recovery through a non-blind deconvolution process suited to non-static blurs. The first step of the algorithm aims at retrieve the motion which the camera undertook during exposure provided only the blurry image. This involves a prediction phase, to lower the blur effect and improve image quality bilateral filtering, a blur parameter estimation phase to identify camera motion features which correctly describe the blurred image from the previous step and a latent image estimation using non-blind deconvolution (Hirsch et al, 2011). Blur parameters are approximated by finding the most appropriate non-static blur which converts the immediate image estimate into the recorded blurry image. Sharp image updates are applied repeatedly during the blur estimation phase. After the blur parameters have been approximated and set, the final sharp image is reconstructed by replacing the image prior of the sharp image update phase with a more natural image prior which is formed on sparsity of gradient images (Hirsch et al, 2011). stationary non-blind deconvolution is then used in the non-stationary case to retrieve the sharp image.



## **2.2 Dedicated camera installed at entrance to indoor car park**

Similar to the CCTV camera located at the side of the road, this system is affected by certain factors which may affect the quality of the images captured by the camera although not as much as an outdoor system. For instance, location at which the camera is placed can be controlled and hence it is placed at the best location possible (that is, the entrance of the car park).

### **2.2.1 Noise Removal**

Noisy images might be a problem that may be experienced with images taken in camera located in the indoor car park. This noise occurs as a result of arbitrary variances in image intensities which appear as grains in the image (Kaur, 2015) . Noise is undesirable because it distorts the image signal and reduces the image quality. Kaur (2015) explains some primary sources of noise in images. These include poor light levels and the temperature of the sensor, electromagnetic transmission of the image data, effects of the environment on the image sensor, introduction of noise by the mechanism capturing image data in digital format, dust particles present on scanner screen. Noise can be removed using various techniques however the technique explained in this report is filtering (linear and non-linear).

Linear filters eliminate certain types of noise by convolving the noisy image with a kernel which serves as a low pass filter (Kaur, 2015). Non-linear filtering on the other hand, produces an output which is not a linear function of its input. Using a gaussian filter to remove noise will be a viable solution however, the problem with using a gaussian filter is that, although it removes the noise, it also blurs the sharp edges and destroys useful details within the image. Therefore, using non-linear filtering is a much better solution because, non-linear filters preserve the image details in the image. For this scenario

the preferred filtering technique is using a median filter. Median filters work by using a sliding window principle over the image matrix and replaces pixel values with the median value of neighbouring values. Initially, all pixels values are arranged from neighbourhood to numerical order. The pixel being considered the middle pixel value is then replaced (Kaur, 2015). Moreover, the median filter although simple, is very powerful and easy to implement.

### 2.2.2 Improving Low resolution images

Assuming that the camera being used captures low resolution images, it is essential that the quality of the images are improved before being passed to the next phase. Similar to noise removal, there are various methods of improving the image quality. however in this scenario, Freeman et al (2002) present a method of improving image quality using super-resolution. Their approach involves using interpolation algorithms to generate tenable high frequency details in zoomed images from a database of training images. This can be used to enhance the quality of low resolution images.

The algorithm works by upscaling the images using cubic spline interpolation and downscaling using convolution with a  $\begin{bmatrix} 0.25 & 0.5 & 0.25 \end{bmatrix}$  filter and subsampling when indices obtained are even. The super-resolution algorithm then anticipates the next octave up that is, the lost frequencies from the image using cubic interpolation. The result the algorithm produces is total of the inputs and high frequency estimates. The high frequencies are estimated for  $M \times N$  patches in raster scan order (Freeman et al, 2002). The algorithm functions under the presumption that, the association between high and low frequency patches is not affected by contrast, hence each patch is normalised by the median absolute value of the low frequency patches in the colour channels. These frequency patches are then merged together to form a vector that is used to search for matches in the training set. The matching is accomplished by locating the closest neighbour. When matches are located, the contrast normalisation is undone on the matching high frequency patch

and that patch is added to the output image.

### 2.2.3 Removal of Barrelling Effect

According to Gribbon et al (2003), The barrelling effect also known as barrel distortion is a phenomenon which happens when the magnification of the lens reduces with axial distance and hence, resulting in rapid movement of image points in the direction of the image centre. Gribbon et al (2003) define the barrel distortion model as:

$$r_u = r_d(I + kr_d^2).$$

Where  $r_d$  and  $r_u$  represent the lengths from the centre of the distorted and undistorted images. The barrel distortion algorithm works by calculating the position of the pixel within the barrelled image. After computation, the resulting coordinates are seldom integers. This problem is solved using bilinear interpolation. The barrel model presents the pixel positions (coordinates) of the undistorted image as a function of the pixel positions of the distorted image, however Gribbon et al (2003) argue that, this representation is not suitable for correcting the distortion in real time, because, the output of the undistorted pixels must be in raster scan order. Therefore, the coordinates of the image without barrelling must be used to decide the pixel to display in the distorted image. Hence the first equation is rewritten in terms of a radial independent magnification with a magnification factor  $M$ . This can be seen below:

$$M = \frac{1}{1 + kM^2r_d^2}$$

The equation above is used to solve for  $M$  repetitively until  $M$  moves or converges towards the required precision (Gribbon et al, 2003). Bilinear interpolation was implemented by shortening the computed coordinates by rejecting the fractional element. This leads to rounding errors in the pixel location and hence produces artefacts. The algorithm derives the pixel values

by computing the weighted sum of pixel values of four nearest neighbours surrounding the computed coordinates (Gribbon et al, 2003).

#### 2.2.4 Contrast Stretching

Images captured from the entrance of the car park may contain night glare caused by the light emerging out of the headlights of the car. This glare effect is undesirable because, it can result in an image having poor contrast. A solution to this problem is to use contrast stretching. Contrast stretching aims to enhance the contrast by expanding the range of intensity values within the image across the range of pixels allowed by the input image (Fisher et al, 2003). To perform contrast stretching first, the lower and upper pixel limits which determine the pixel range for the normalisation process must be defined. The image is then scanned to identify the highest and lowest pixel values within the image. Suppose that the lower and upper pixel limits are  $i$  and  $j$  respectively and the highest and lowest pixel values are  $k$  and  $l$  respectively, it is possible to scale each pixel using the following equation (Fisher et al, 2003):

$$P_{out} = (P_{in} - l) \frac{j-i}{k-l} + a$$

This function faces a problem where outlying pixels that is, pixels with either very low or very high intensities can lead to unnatural scaling. As a result, a more efficient approach will be to use the histogram of the image to obtain the  $k$  and  $l$  values approximately at the 5th and 95th percentile as this stops outlying pixels from over-scaling (Fisher et al, 2003). Alternatively, outliers could be dealt with by using intensity histograms to locate the most common pixel intensity values and then defining a threshold value at which any value below the threshold will be neglected. After this the intensity histogram is traversed from zero to the first intensity value which is above the defined threshold value. This value is then set to  $l$  Furthermore, the intensity histogram is then traversed downwards from 255 till the first intensity value

above the threshold value. This value is then set to  $k$ .

### 3 The proposed VMMR system

This section explains and discusses the proposed vehicle make and model system and the steps involved in the design of the system.

#### 3.1 High level block diagram



Figure 1: High level block diagram

The following sub-sections discuss the function of each block in the block diagram.

##### 3.1.1 Loaded images

The first module of the system represents all the images that are loaded from the database. A datastore was created to hold the images within the subdirectories so as to ensure that all the image data is held centrally for easy access during the implementation phase.

##### 3.1.2 Feature Detection

It is at this stage that features or key points are detected within the image using a feature detector algorithm. These features include, edges, lines, blobs, corner points etc. The number of features detected depends on the input parameters passed to the feature detection algorithm.

### **3.1.3 Feature Extraction**

This is where the features or key points are extracted from the images based on the features detected by the feature detection algorithm. The feature extraction algorithm produces a feature vector containing the extracted features from a local bounding box within the image.

### **3.1.4 Creation of Bag of Features**

The bag of SURF features is created within this module after the features have been extracted by the feature extraction algorithm. K-means clustering is used to create a visual word vocabulary from the feature vectors obtained from the previous module.

### **3.1.5 Training**

This is where a classifier, in this case, a support vector machine (SVM) is trained using the bag of SURF features and the corresponding labels.

### **3.1.6 Prediction of classes (Testing)**

Within this module, trained SVM attempts to predict the make and model of a car from an image within the validation or test set.

### **3.1.7 Results**

This is where the predictions made by the support vector machine is evaluated against the actual outcome.

## **3.2 System Design**

### **3.2.1 Loading the images**

The frontal view of various makes and models were captured and stored in the database. The images were placed into subdirectories according to the name of the make and model of each car. These images were preprocessed before they were loaded into the system. This was done in order to ensure that the program runs quicker and is less expensive computationally than preprocessing the images at run time. The preprocessing tasks that were carried out were resizing all images to ensure uniformity within the images and converting the images from the *RGB* colour space to grayscale. Moreover, the image datastore was split into two sets that is, a training set and a validation set. Each subdirectory had a total of 25 images, therefore, it was decided to use 20 images for training and 5 images for testing. Hence 500 images were used for training and 125 images were used for testing.

### **3.2.2 Feature detection algorithm**

The bag of features approach was used along with Speeded Up Robust Features (SURF) to create a bag of SURF features. Local SURF features were selected using the grid approach.

### **3.2.3 Extraction of SURF features**

The SURF features were extracted using an 8 by 8 patch and a bounding box which used the following coordinates [32 64 96 128] (MathWorks, 2017).

### 3.2.4 Creation of Bag of Features

The bag of features was created from the extracted SURF features. After extraction, 80 percent of the strongest features also referred to as codewords are retained and the rest of the features are discarded. The number of features are then balanced across all categories as to improve the efficiency of the k-means clustering (MathWorks, 2017). To balance the features, the algorithm locates the image from which the least amount of features were obtained within a category and then selects the strongest amount of features from all categories based on this amount (MathWorks, 2017), therefore, the strongest amount of features selected throughout the images are the same. K-means clustering is used to generate a visual vocabulary of 500 words, hence a total of 500 clusters are created using the features.



## 4 Experiments and Results

## 5 Reference List

- Cho, T., S. (2010). *Motion blur removal from photographs*. Submitted to the Department of Electrical Engineering and Computer Science in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical Engineering and Computer Science. Cambridge Massachusetts: Massachusetts Institute of Technology.
- Freeman, W., J., T., Jones, T., R., Pasztor, E., C. (2002). Example-based super-resolution. *IEEE Computer Graphics and Applications*, [Online] 22(2), pp.56-65. Available from: <http://ieeexplore.ieee.org/document/988747/> [Accessed 6 Jan. 2018].
- Fisher, R., Perkins, S., Walker, A., Wolfart, E. (2003). *Contrast Stretching*. [Online]. Available from: <https://homepages.inf.ed.ac.uk/rbf/HIPR2/stretch.htm> [Accessed 7th January 2018].
- Gonzalez, C., R., Woods, E., R. (2008). *Digital Image Processing*. Upper Saddle River New Jersey: Pearson.
- Gribbon, K. T., Johnston, C. T., and Bailey, D. G. *A Real-time FPGA Implementation of a Lens Distortion Correction Algorithm with Bilinear Interpolation*. Proceedings of the Image and Vision Computing New Zealand Conference 2003, Massey University, Palmerston North, New Zealand, Nov. 2003. pp. 408-413,
- Hirsch, M., Schuler, C., Harmeling, S. and Scholkopf, B. (2011). Fast removal of non-uniform camera shake. 2011 International Conference on Computer Vision. [online] Available from: <http://ieeexplore.ieee.org/document/6126276/> [Accessed 5 Jan. 2018].
- Kaur, S. (2015). Noise Types and Various Removal Techniques. *International Journal of Advanced Research in Electronics and Communication En-*

*gineering (IJARECE)*. 4(2). [Online] p.226-229. Available from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.683.6783&rep=rep1&type=pdf> [Accessed 5th January 2018].

MathWorks (2017). *Image Category Classification Using Bag of Features*. [Online]. Available from: <https://uk.mathworks.com/help/vision/examples/image-category-classification-using-bag-of-features.html> [Accessed: 8th January 2018].

Patel, J., Shah, A., Patel, H. (2015). Skew Angle Detection and Correction using Radon Transform. *International Journal of Electronics, Electrical and Computational System*. [Online] Volume 4 (May) Available from: <http://academicscience.co.in/admin/resources/project/paper/f201505101431314896.pdf>. [Accessed: 2nd January 2018].

Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. New York: Springer.

Weng, C., C. Homer, C. Fuh, C., S. A Novel Automatic White Balance Method For Digital Still Cameras. 2005 *IEEE International Symposium on Circuits and Systems*. [Online] Available from: <http://ieeexplore.ieee.org/abstract/document/1465458/> [Accessed 4 Jan. 2018].