An Improved CAMShift Algorithm for Object Detection and Extraction

Abdulmalik Danlami Mohammed* and Tim Morris
School of Computer Science
University of Manchester
Oxford Road, Manchester, UK
abdulmalik.mohammed@postgrad.manchester.ac.uk, tim.morris@manchester.ac.uk

Abstract

Continuously Adaptive MeanShift (CAMShift) is an important algorithm for object tracking based on the colour histogram. The algorithm works by finding the mode of a probability distribution map within a search window and iteratively updates the position and size of the window until convergence. The algorithm boasts of high performance in a simple environment where the colour distribution is constant. However, because the algorithm is dependent on a static colour distribution, its performance suffers in cases where the distribution changes e.g. due to illumination or weather conditions. In addition, object occlusion and complex background colour can degrade the performance of the algorithm. In this paper, we propose a CAMShift algorithm that can track coloured signs. Since multiple colours are involved for tracking, we utilized a Bayesian approach to estimate the colour probability density function. This probability density function gives the probability of whether a pixel value corresponds to certain object. We illustrate the effectiveness of our approach by detecting and extracting visual sign images with different colour attributes. The result obtained shows that our extended CAMShift algorithm can detect and track coloured signs based on the identified colour class.

Keywords: CAMShift algorithm, object tracking, colour histogram

1 Introduction

The CAMShift algorithm was originally developed for a head and face-tracking application [12] based on the colour histogram of the target object. It was an improvement over the traditional mean shift algorithm that finds the peak of a probability distribution generated from a colour histogram. The algorithm is used as an important part of other applications such as video surveillance, human computer interaction, head and face tracking applications [15]. The basic workflow of CAMShift is illustrated in Figure 1. The location and size of the search window is initialized to include the area that contains the target object in a captured image. The hue value for each pixel within the search window is sampled to generate the probability density function. This density function is saved as the histogram model of the target object. The next step is to estimate the probability distribution map. The histogram model scans through each pixel of the captured scene to determine the probability of that pixel belonging to the target object.

The main part of the CAMShift algorithm is the area within the dotted lines of Figure 1. The following steps summarize the operation of CAMShift:

S1. Set (x, y) as the initial location of the search window

IT CoNvergence PRActice (INPRA), volume: 2, number: 2, pp. 55-65

*Corresponding author: Advance Interface Group(AIG), School of Computer Science, University of Manchester, Room 2.94 Kilburn Building, Oxford Road. M13 9PL, Manchester, United Kingdom, Tel:+447741052107

S2. Find the zeroth, first and second order image moments of the search window which are given below:

$$M_{00} = \sum_x \sum_y I(x,y). \tag{1}$$

$$M_{10} = \sum_x \sum_y xI(x,y); M_{01} = \sum_x \sum_y yI(x,y); \tag{2}$$

$$M_{20} = \sum_x \sum_y x^2 I(x,y); M_{02} = \sum_x \sum_y y^2 I(x,y); \tag{3}$$

Where I(x,y) is the pixel (probability) value at position (x,y) in the image, and x and y range over the search window.

S3. Then the mean location (centroid) in the search window is computed as:

$$x_c = \frac{M_{10}}{M_{00}}; y_c = \frac{M_{01}}{M_{00}} \tag{4}$$

To update the search window, the target object aspect ratio, given below:

$$ratio = \frac{M_{02}}{y_c^1} / \frac{M_{20}}{x_c^1} \tag{5}$$

is used with:

$$width = 2M_{00}.ratio; height = \frac{2m_{00}}{ratio} \tag{6}$$

S4. Repeat Steps 2 and 3 until convergence, i.e. the parameters do not change

In recent years, CAMShift has attracted a lot of research interest in the computer vision and image processing domains because of its simplicity and effectiveness for object tracking applications. Many of these researches are concerned with improving the algorithm's performance to cope with challenges militating against object tracking applications. While the original implementation of the algorithm utilized the hue component of the HSV colour space for a head and face tracking application [12] in a constrained environment, the algorithm is vulnerable to tracking failure in environments with a complex background or where the value of hue varies due to lighting changes. Object occlusion is another challenge that can cause tracking failure. Furthermore, a background with similar colouration to the target object can result in false detection and thus affect the accuracy of CAMShift. In this paper, we propose an extension to CAMShift for tracking an object based on identified colour properties. The proposed approach addresses the issue of lighting variations. The contribution of this paper is two fold: 1) the tracking of different coloured signs with CAMShift for object tracking, 2) the significance of the prior probability in classification accuracy is investigated. This work is part of an ongoing PhD project in Image processing on Mobile Platform. As part of the theme of IT convergence, this work has a practical application in autonomous navigation using smart phone. The remaining parts of this paper are organised as follows. In Section 2, we discuss other works as they relate to CAMShift. Section 3 introduces a Bayes' classifier and its application in 2D histogram normalization. In Section 4, we present the extended CAMShift algorithm. While Section 5 discusses the results of tracking visual signs using our proposed method, Section 6 concludes the discussion of our proposed object tracking technique.

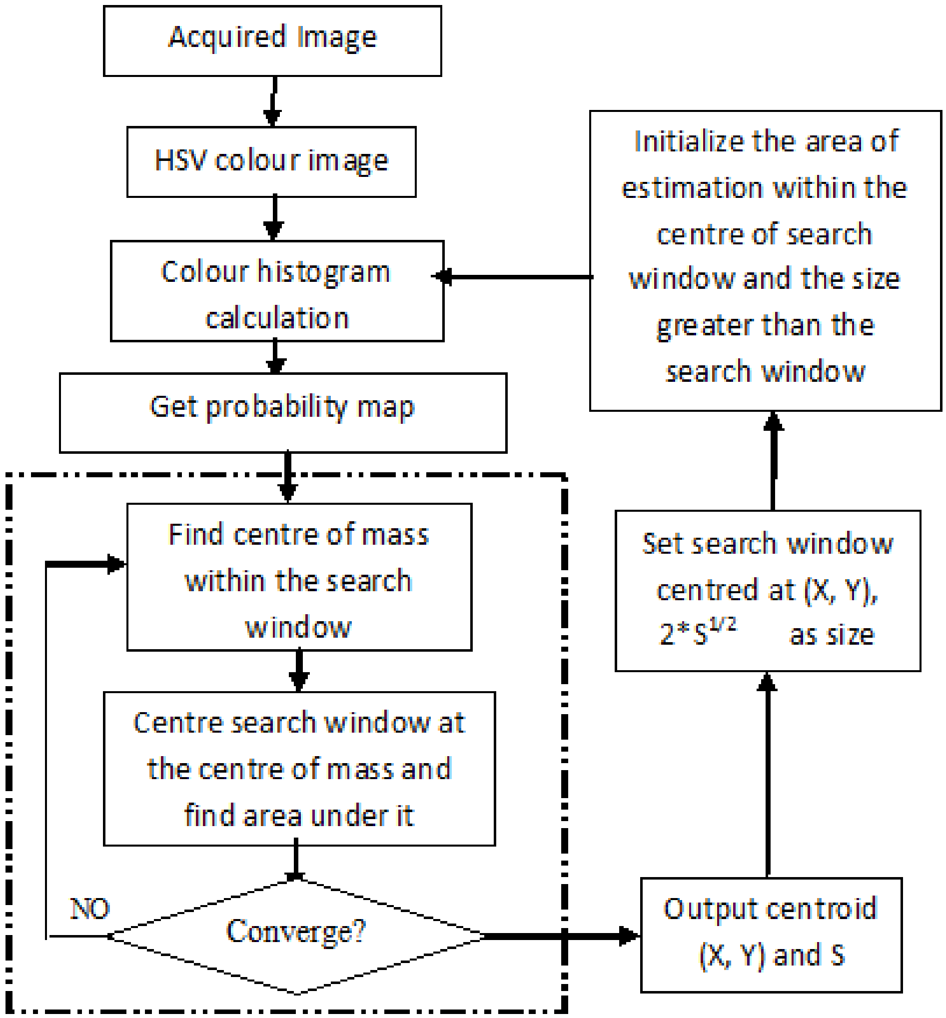

Figure 1: Object Tracking Schema

2 Related Works

This section provides a detailed review of other works as they relate to object tracking using the CAMShift algorithm. In addition, the section discusses the merits and shortcomings of the many different features of a target object that are employed in different object tracking applications.

Several tracking approaches have been proposed in the literature. While most of these studies have used colour information as an object feature for tracking, other approaches integrate this information with other features such as edge, texture, object geometry and motion characteristics of the target object. For example, to overcome the challenge of object occlusion during tracking, CAMShift based on colour and texture features is proposed in Alper et al. [3] for tracking. In addition to the probability distribution, the shape information of the target object is used as a feature. The integration of the shape information is to re-establish object tracking in the case of tracking failure due to occlusion. A similar approach that utilises colour features and shape characteristics of the target object for tracking is presented in [17]. This approach utilises skin colour and contour information of the target object in a modified CAMShift algorithm. The use of contour information has become crucial since the target object to track has a constant outline; hence the algorithm fitted a contour, for instance an ellipse, to the ROI to improve the location of the target. While these approaches promise robust object tracking against occlusion, they still suffer drawbacks especially when the object is set against a complex background, where the colour property of the object and background might be similar.

To overcome the issue of occlusion and thus improve object tracking, Donghe and Jinson [14] introduced a CAMShift tracking algorithm based on parameter prediction filters. The algorithm works by locating the target in the first three frames of the video sequence and using this position to initialise the parameters of a prediction filter to find the location of the target in the following frames. While it is robust against occlusion, it fails to address the issue of image brightness as occasioned by varying illumination, which can lead to a tracking failure. A visual tracking approach based on an adaptive colour histogram model is proposed in [4]. This approach is designed to be robust against lighting changes through an online adjustment of the object parameters. In the case of this approach, object tracking may fail due to occlusion.

A frame differencing method for object detection and a CAMShift algorithm based on integration of multiple features such as hue and edge orientation histogram of the target object for tracking is proposed in [18]. In the face of a complex background, this algorithm will fail to track targets.

A kernel based object tracking using CAMShift is proposed in [1]. Prior to histogram back-projection, the multidimensional histogram is weighted with a decreasing kernel profile. Even though the effectiveness of the algorithm against complex background is reported, it failed to address other challenges facing object-tracking applications such as occlusion, varying image brightness, image blurring and clutter. In contrast to [1], [7] proposed an object tracking approach based on background weighted kernel to address the issue of partial occlusion. Another kernel based object tracking algorithm is introduced in [5]. In this work, the histogram representation of the target object is normalised with an isotropic kernel to generate a similarity function that enables the localization of the target object.

A head and face tracking approach in a perceptual user interface using hue of the HSV colour space is presented by Bradski [12]. The algorithm is simple and robust in a controlled environment. However, it suffers drawbacks in the presence of occlusion, complex background, and when the hue value changes due to illumination changes. A similar approach is proposed in [11], though with some modifications. To overcome the lighting issue Exner et al. [17] make use of accumulated histograms of the target object to represent varying appearances.

In order to deal with the issue of complex background, [2] modified the CAMShift algorithm by modelling motion features of the target object and concatenating them with colour features. The result of this union is a probability distribution. The algorithm was implemented for a human-machine interaction application. The experimental results show that the algorithm is robust to illumination changes, partial occlusion and complex background. A similar work that addresses the issue of similar background and object colouration is reported in [13]. In this paper, a CAMShift algorithm based on background model is used for object tracking. The algorithm is able to adapt to a changing environment because of the online learning scheme of the background model over time. This online learning scheme can add computational burden to the algorithm.

An object tracking approach that employs an online learning scheme of object appearance is proposed in [6]. While the algorithm may be robust to lighting changes, it may fail to track an object that is covered or occluded by other objects.

In [9], a CAMShift algorithm based on the colour histogram is employed for tracking traffic lights. The algorithm is implemented as part of traffic light recognition system. As with other colour based tracking methods, this algorithm suffers from occlusion and may fail to track the traffic light when the background colour corresponds to a traffic light colour. A similar algorithm that utilises colour histogram and histogram back projection is presented in [16]. The algorithm is implemented for human face tracking system.

Recent research has seen CAMShift being implemented in mobile devices. For example, Mohammed and Morris [11] implemented an object tacking method using CAMShift algorithm based on the colour histogram on a camera phone. The approach utilises multiple histograms representing object appearances under different illuminations to address the problem of lighting changes. While this approach is robust to illumination changes, it can be vulnerable to tracking failure due to object occlusion. Another object tracking approach on a mobile phone is proposed in [19]. The work utilises local image features and updated those features online, in order to address the variation in object appearances.

In summary, the use of features for object tracking is application dependent, some algorithms are robust in certain implementation others require the integration of other features at the expense of additional computation to achieve their objective.

3 Bayesian approach

This section gives an overview of the Bayes classifier and its application in normalizing 2D histograms. In particular, the section describes the influence of prior probability on classification accuracy.

The Bayes classifier is a classification technique that makes use of statistical measures of trained data to group individual observations into different classes. This definition can be expressed as shown in equation 7.

$$p(w_i|x) = \frac{p(x|w_i)p(x_i)}{p(x|w_i)p(x_i) + p(x|w_j)p(x_j)} \tag{7}$$

Where $p(w_i|x)$ represents the posterior probability, whose value determines the likelihood of an object x belonging to a class w_i. Given two classes of patterns w_i and w_j, an object x is said to belong to class w_i, if $p(w_i|x)$ is greater than $p(w_j|x)$. That is:

$$x \in w_i, if\, p(w_i|x) > p(w_j|x) \tag{8}$$

In the case of a captured image, the classifier determines the background and foreground classes of individual pixels based on the value of the posterior probability. For example, a pixel is classified as a background if the posterior probability is less than a certain threshold otherwise it is classified as foreground.

$$p(w_i|x) = \left\{ \begin{array}{ll} 1, & \text{Foreground if } p(w_i|x) > T \\ 0, & \text{Background if } p(w_i|x) \leq T \end{array} \right\} \tag{9}$$

$p(x|w_i)$ and $p(x|w_j)$ represent the class conditional states. We consider a normalised 2D histogram in our approach to represent the class density function. The 2D histogram is made up of the Hue and Saturation channels of the HSV colour space. Figure 3 shows 2D and 3D representations of the distribution of hue and saturation. Further discussion of the estimation of $p(x|w_i)$ can be found in section 4.

$p(x_i)$ and $p(x_j)$ represent prior probabilities for foreground and background classes. Often this probability is assumed, based on the number of classes involved in the classification. For instance in [8], the prior probability is assumed equal for each of the five classes involved. Even though many approaches give little or no attention to the effect of prior probability on classification accuracy, it is observed that a little change in the value of a prior probability can influence the accuracy of a classifier. While evaluating classifier performance, it is important to pay attention to the values of prior probability in order to achieve a correct result. Table 1 shows the accuracy of our classification technique using different values of prior probability. As can be seen from the table, column 1 depicts the different values of the prior that is used to estimate the probability of a pair of hue and saturation belonging to specific class.

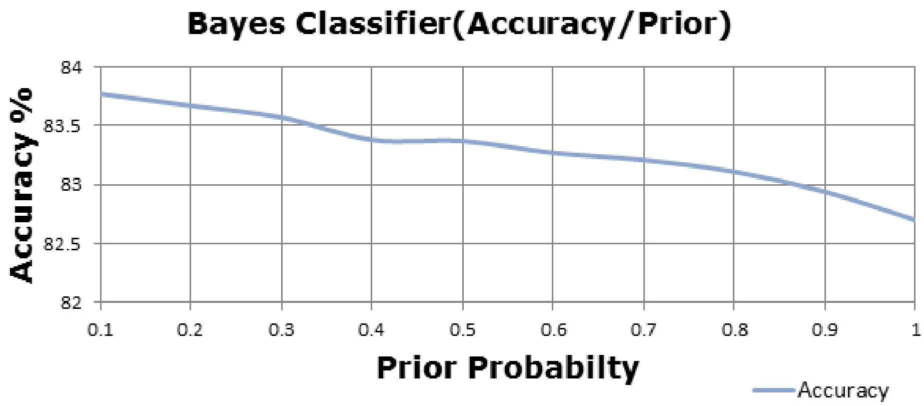

Figure 2: Influence of Prior Probability on the Accuracy of Bayes Classifier

This probability is computed using equation 7. In our experiment, 46086 pixels from 125 images were processed to extract the hue and saturation value. These extracted values together with their class label represent our training data. The label classes are estimated for 10 different values of a prior and for each of the value, we trained a classifier. The accuracy of our classifier during testing is shown in column 4 of Table 1. In Figure 2, we plot a graph of accuracy against prior probability to give pictorial information about the influence of prior probability on the accuracy of classifier.

Prior Probability	TP	FN	Accuracy
0.1	80.7	0.92	83.77
0.2	80.5	0.94	83.67
0.3	80.3	0.95	83.57
0.4	79.9	0.99	83.38
0.5	79.8	0.98	83.37
0.6	79.7	0.99	83.27
0.7	79.5	0.98	83.21
0.8	79.4	0.98	83.11
0.9	79.2	0.96	82.94
1.0	79.0	0.90	82.70

Table 1: Accuracy of Bayes classifier based on Prior Probability

As shown in Figure 2, the prior probability has little or no effect on the accuracy of a classifier. We have observed in the course of our experiment that for simple coloured images where the distribution of foreground and background colour are separable, the prior probability can assumed any value without affecting the accuracy of classifier.

4 Extended CAMShift Algorithm

Visual objects are known to have colour attributes that define their state and thus make them stand out from other objects in the scene. For example, a stop sign has a red background colour, a warning or caution sign can be identified with a yellow colour. Hence, colour can be a helpful feature for object detection and tracking.

Several colour models such as RGB, HSV, YUV, etc are diversely used for colour processing applications. Most of these colour models are application dependent [10]. For example, RGB is implemented in hardware, while the HSV has practical application in software development. The dimensions of the HSV colour space are Hue (corresponding to the basic colour), Saturation (the depth of the colour) and Brightness. The separation of chromaticity (coded in the H and S channels) and brightness enables us to define colour ranges that are invariant to moderate illumination changes and thus colour selection is made possible. In this work therefore, we utilize the Hue and Saturation channels of HSV colour space to build a histogram of target colours.

Our proposed work is an extension to the work presented in [11], but in contrast to [11], we have extended the CAMShift algorithm to track other coloured signs. For example, signs whose background colours are red or yellow can be detected and tracked using our method. The proposed technique is the same as in [11] except for the estimation of the model histogram, hence in this paper, we only discuss the estimation of the model histogram based on the Bayesian approach. Figure 3 shows the colour distribution of our data set in 2D and 3D graphs. For a comprehensive discussion about data analysis and histogram accumulation, the interested reader is referred to section 3 of the work presented in [11].

4.1 Model Histogram

To compute our model histogram we use Equation 1. This can be expressed informally as:

$$model\_hist = \frac{hist(GREEN+RED+YELLOW)\times prior}{(hist(GREEN+RED+YELLOW)\times prior)+hist(NONGRY*(1-prior)} \tag{10}$$

Where:

model_hist Defines the model histogram that is the probability of a given pixel with value x belonging to green, red or yellow colour.

hist(GREEN+RED+YELLOW) This represents a 2D histogram based on Hue and Saturation (H-S) value of HSV colour space derived from positive samples of the targets to be identified. The histogram is generated from images in different shades of green, red and yellow colours. It is the probability that a pixel from an image has green, red or yellow value. As part of our data set, it represents the positive class (of which there were 75 images). We improve accuracy by excluding any white or black text or icons from this set.

prior The prior probability as explained in section 3 can affect the outcome of a classifier hence, in choosing the right value for the prior, factors such as the number of positive classes involved have to be considered. In our case, we assume one as the value of a prior probability.

hist(NONGRY) Is a 2D histogram that is calculated in the same way as **hist**(**GREEN** + **RED** + **YELLOW**) except that different coloured images (not red, green, yellow) in different shades are used. As part of the data set, it represents the negative class (50 images in our experiment).

5 Experimental Result

The result of applying our extended CAMShift algorithm on visual signs is shown in Figure 4. Row 1 of Figure 4 shows the original images of different signs captured using an Android camera phone, row

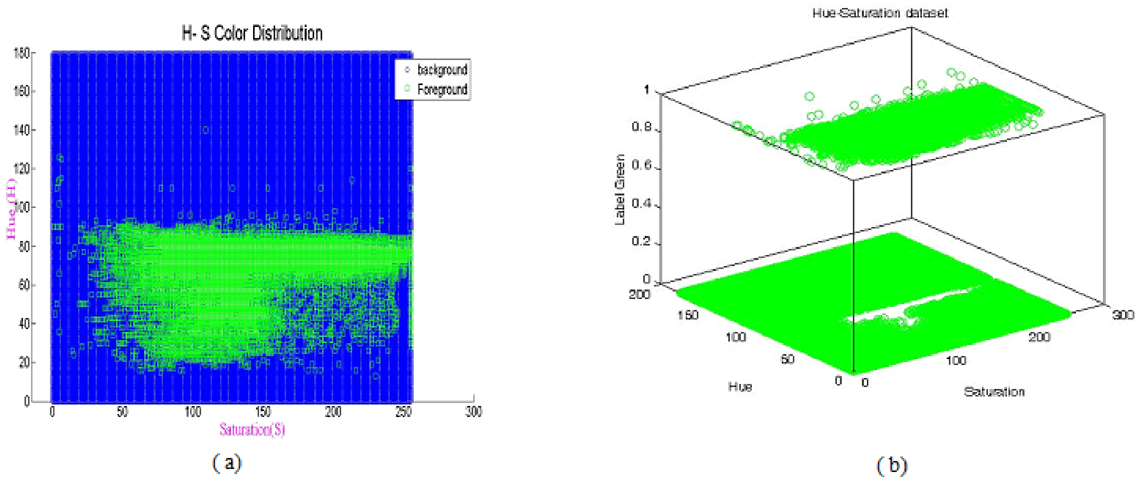

Figure 3: (a) 2D representation of the H-S colour distributions; (b) 3D Histogram of H-S colour distributions

2 shows the result obtained using our extended CAMShift algorithm. In row 2 the detected areas of the images are bounded by a red rectangle.

Figure 5 depicts the extracted signs. These extracted regions can be helpful for further object analysis in applications such as object recognition.

6 Conclusion

In this paper, we have extended a CAMShift Algorithm to track objects of different colour content. The approach is thus an extension to initial work proposed in [11]. We utilized a Bayesian approach to estimate the 2D histogram that comprises green, red and yellow colours. This estimation is done offline to speed up processing at run time.

As can be seen from row 2 of Figure 4, our approach addresses the problem of object detection (and tracking) when the object appearance changes as a result of weather condition. While the object tracking technique implemented here is simple and effective, it can fail in situations where the object is occluded. Hence, our future work will integrate shape features with colour to achieve a robust tracking in face of lighting changes and occlusion.

References

[1] John G. Allen, Richard Y. D. Xu, and Jesse S. Jin. A cooperative intrusion detection system for ad hoc networks. In *Proc. of the Pan-Sydney area workshop on Visual information processing (VIP'05), Hong Kong*, pages 3–7. ACM, December 2005.

[2] Boris Babenko, Ming-Hsuan Yang, and Serge Belongie. Robust object tracking with online multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1619–1632, August 2011.

[3] Gary R. Bradski and Santa Clara. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal(2nd Quarter)*, 2(2):1–15, 1998.

[4] Tse Min Chen, Ren C. Luo, and Tsu Hung Hsiao. Visual tracking using adaptive color histogram model. In *Proc. of the 25th Annual Conference of the IEEE Industrial Electronics Society (IECON'99), San Jose, USA*, pages 1336–1341. IEEE, November-December 1999.

Figure 4: Row 1: Images of coloured signs; Row 2: Detected area of the image.

Figure 5: Extracted regions of signs

[5] Dorin Comaniciu, Visvanathan Ramesh, and Peter Meer. Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):564–577, May 2003.

[6] David Exner, Erich Bruns, Daniel Kurz, and Anselm Grundh

[7] Lixin Fan, Mikko Riihimaki, and Iivari Kunttu. A feature-based object tracking approach for realtime image processing on mobile devices. In *Proc. of the 17th IEEE International Conference on Image Processing (ICIP'10), Hong Kong*, pages 3921–3924. IEEE, September 2010.

[8] Aly A. Farag and Alaa E. Abdel-Hakim. Detection, categorization and recognition of road signs for autonomous navigation. In *Proc. of the Advanced Concepts for Intelligent Vision Systems (ACIVS'04), Brussels,*

Belgium, pages 125–130, September 2004.

[9] Jianwei Gong, Yanhua Jiang, Guangming Xiong, Chaohua Guan, Gang Tao, and Huiyan Chen. The recognition and tracking of traffic lights based on colour segmentation and camshif for intelligent vehicles. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV'10), San Diego, USA*, pages 431–435. IEEE, June 2010.

[10] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Prentice Hall, 2002.

[11] Jaideep Jeyakara, R. Venkatesh Babub, and K.R. Ramakrishnanc. Robust object tracking with background-weighted local kernels. *Journal of Computer Vision and Image Understanding*, 112(3):296–309, 2008.

[12] Abdulmalik Danlami Mohammed and Tim Morris. A robust visual object tracking approach on a mobile device. In *Proc. of the 2nd International Conference on Information and Communication Technology (ICT-EurAsia'14), Bali, Indonesia, LNCS*, volume 8407, pages 190–198. Springer-Verlag, Apri 2014.

[13] R. Stolkin, I. Florescu, and G. Kamberov. An adaptive background model for camshift tracking with a moving camera. In *Proc. of The 6th International Conference on Advances in Pattern Recognition (ICAPR'07), Kolkata, India*, pages 147–151, January 2007.

[14] Donghe Yang and Jinsong Xia. Face tracking based on camshift algorithm and motion prediction. In *Proc. of the International Workshop on Intelligent Systems and Applications (ISA'09), Wuhan, China*, pages 1–4. IEEE, May 2009.

[15] Alper Yilmaz, Omar Javed, and Mubarak Shah. Object tracking: A survey. *Journal of ACM Computing Surveys*, 38(4):1–45, 2006.

[16] Tae-Woong Yooa and Il-Seok Oh. A fast algorithm for tracking human faces based on chromatic histograms. *Journal of Pattern Recognition Letters*, 20(10):967–978, October 1999.

[17] LI YUAN and ZHI-CHUN MU. Ear detection based on skin-colour and contour information. In *Proc. of the 6th International Conference on Machine Learning and Cybernetics (MLC'07), Hong Kong*, pages 2213–2217. IEEE, August 2010.

[18] Yingying YUE, Yun GAO, and Xuejie ZHANG. The recognition and tracking of traffic lights based on colour segmentation and camshif for intelligent vehicles. In *Proc. of the 1st International Conference on Information Science and Engineering (ICISE'09), Nanjing, China*, pages 1141–1144. IEEE, December 2009.

[19] Tengyue Zou, Xiaoqi Tang, and Bao Song. Improved camshift tracking algorithm based on silhouette moving detection. In *Proc. of the 3rd Third International Conference on Multimedia Information Networking and Security (MINES'11), Shanghai, China*, pages 11–15. IEEE, November 2011.

Author Biography

Abdulmalik Danlami Mohammed is a member of Nigeria Computer Society (NCS) and International Association of Engineering (IAENG). He holds a B.Sc degree in Computer Science, which was obtained from Saint Petersburg State Electro-Technical University (2003), Russia and M.Sc Degree in the same field from Belarussian National Technical University (2010), Belarus. He is a PhD research scholar at the University of Manchester, UK. His research interest includes Image processing, Computer Vision, Machine learning, Mobile Computing and Computer Networks.

Tim Morris is a lecturer at the school of Computer Science, University of Manchester. He gained BSc and PhD degrees in Physics and Medical Physics from the Universities of Southampton and Sheffield respectively. Apart from the academic papers listed separately, he has published two textbooks: Multimedia Systems (Springer-Verlag, 2000) and Computer Vision and Image Processing (Palgrave, 2004). He is a member of the IEEE and a member of their Technical Committees on Multimedia Computing, Pattern Analysis and Machine Intelligence, and Emergent Technologies.