Carried object detection in videos using color information

This item was submitted to Loughborough University’s Institutional Repository by the/an author.


Additional Information:

- This article was published in the journal IEEE Transactions on Information Forensics and Security (© 2013 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other users, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted components of this work in other works).

Metadata Record: https://dspace.lboro.ac.uk/2134/13745

Version: Accepted for publication

Publisher: © IEEE

Please cite the published version.
This item was submitted to Loughborough’s Institutional Repository (https://dspace.lboro.ac.uk/) by the author and is made available under the following Creative Commons Licence conditions.

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/
Carried Object Detection in Videos using Colour Information

G. Tzanidou, I. Zafar, E.A. Edirisinghe,

Department of Computer Science, Loughborough University, Loughborough, UK

Abstract—Automatic baggage detection has become a subject of significant practical interest in recent years. In this paper we propose an approach to baggage detection in CCTV video footage that uses colour information to address some of the vital shortcomings of state-of-the-art algorithms. The proposed approach consists of typical steps used in baggage detection namely: the estimation of moving direction of humans carrying baggage, construction of human-like temporal templates and their alignment with the best matched view-specific exemplars. In addition we utilize the colour information to define the region that most likely belongs to a human torso in order to reduce the false positive detections. A key novel contribution is person’s viewing direction estimation using machine learning and shoulder shape related features. Further enhancement of baggage detection and segmentation is achieved by exploiting the CIELAB colour space properties. The proposed system has been extensively tested for its effectiveness, at each stage of improvement, on PETS 2006 dataset and additional CCTV video footage captured to cover specific test scenarios. The experimental results suggest that the proposed algorithm is capable of superseding the functional performance of state-of-the-art baggage detection algorithms.

Index Terms—Baggage detection, carried object detection, viewing direction estimation, CIELAB colour space.

I. INTRODUCTION

In recent year the number of cameras installed in public spaces has increased dramatically. A report by K. Ball suggests that the number of surveillance cameras in UK reached 4.2 million, resulting in the impressive analogy of 1 CCTV camera being installed per 14 individuals of the population[2]. This fact brings up the need for automated surveillance and development of security related applications. Often a detailed description of a human appearing in CCTV footage is used in an individual’s behaviour analysis. Detection and recognition of baggage being carried is one of the most widely used descriptions vitally used in theft detection and criminal behaviour identification.

One of the most successful state-of-the-art systems proposed in the area of automatic baggage detection is I. Haritaoglu’s approach presented in [13] and [14], which uses symmetrical attributes of the human body silhouette and periodic motion to detect carried objects. The fundamental idea used in this approach is that anything that violates the symmetry around the human body’s main axes and is thus not a part of the body forms a foreign object. In this approach a temporal textural template is developed over time and a major axis which defines the traversal of the centre of the body is determined by using Principle Component Analysis (PCA). The detected non-symmetric shapes of the template undergo periodicity analysis based on horizontal projection histograms. Javed and Shah in [16] adopted a temporal template to create the Recurrent Motion Image (RMI). In contrast to the aforementioned temporal templates, RMI aims at recording the regions of the moving object that perform kinesis. Subsequently, a symmetry analysis is performed to spot the non-symmetric regions of the silhouette, similar to the approach proposed in [13]. Since the RMI maps the moving parts of the body, a simple comparison of the non-symmetric region with the one in RMI will show if the non-symmetric region belongs to the moving parts or not. In case that that the detected object is non-recurrent, then it is labelled as a carried object.

A further important approach in which baggage detection was based on motion was proposed by C. Ben Abdelkader and L. Davis [3]. If it is taken into account that the manner of walking and the body shape changes when carrying a weight/load, the concept focuses on the analysis of human gait and body shape to determine the presence of carried objects. Therefore, the characteristics (periodicity and amplitude) of naturally walking people are recorded and some constraints, regarding the period of gait and the recurrent motion of arms and legs, are imposed. If any of the constraints is violated then the person is considered to be carrying a bag. A similar approach that uses periodicity information is described by T. Senst et al. in [31], and aims on placing

1 G. Tzanidou, I. Zafar, E. A. Edirisinghe are with the Department of Computer Science, Loughborough University, Loughborough, LE 11 3TU, UK (e-mail: G.Tzanidou@lboro.ac.uk, IZafar@lboro.ac.uk, E.A.Edirisinghe@lboro.ac.uk)
people under two categories, carrying a bag or not. A latest contribution of Y. Ran et al. in [25] that exploits the periodical attribute of human motion to construct helical signatures deals with robust pedestrian segmentation, occlusion handling and carrying status identification.

In [6] and [5] the authors proposed methods based on wavelet analysis and neural networks for detecting intruders carrying objects in archaeological sites. In [6] the Harr wavelet transform was applied to a segmented binary human silhouette. After 3-level decomposition, a sub-band of the last decomposition level is selected and forwarded to a trained, three layer neural network. The output of the neural network processing produces a true or false answer depending on if the moving object is a person or not. An extended edition of work in [6] is [5] where carried object detection is attempted by adding two further trained neural classifiers. The detected, grey-scale human silhouette is scanned using a mask of the carried object in order to detect the item. The work of H. Nanda et al. in [22] is another example of detecting carried objects using neural networks. Each binary image of a moving object is classified as a pedestrian or a pedestrian with distorted shape. The majority of the classification results per subject will indicate the final result. One of the latest contributions which investigate carrying object detection using a classifier is of Qi, Huang and Wang [24]. They based their research on the width of the human silhouette outline in relevance to a human body’s main axis. Support Vector Machine (SVM) classifier was trained with a vector containing the distance of each point on the contour of the binary silhouette from the main axis for all the images selected for training. After the main axis is computed and all the images are resized to the same height, the characteristics of the contour of the silhouette are forwarded to the SVM for classification. Once the person is classified as either carrying an object or not, the location of the object (possibly a bag) is specified. Most of the aforementioned methods, based on classifiers, do not provide information about the position of the bag making them unpopular.

N. M. Ghanem and L. S. Davis [11] approached the topic of baggage detection in terms of finding the changes in the appearance of pedestrians before entering and after leaving a Region of Interest (ROI); for example around a rubbish bin. Similarly, in [9] C.-H. Chuang et al. proposed baggage detection when bag exchange occurs between two individuals via people tracking and calculation of Ratio Histograms (RH). The ratio histogram is a fraction of the colour histograms of each person before and after the bag exchange. The detection of missing colours between the before and after conditions in the Ratio histograms leads consequently to the detection of a carried object and answers to the question “who out of the people involved was carrying the bag before and after the exchange?”.

Two contributions which seems to differ significantly from all the approaches described above, are of C. Lee and A. Elgammal in [19] and of by R. Chayanurak et al. in [8]. The former made use of a novel technique where the body shape of a model is decomposed to body pose, shape style and view, and is reconstructed preserving the pose and varying the shape and the style parameters. The latter utilised the star skeleton that defines the regions representing the limbs and protruding objects. The decision is made by calculating the level of movement that the identified protruding region or limb performs. However, the experimental results presented are inadequate to support this method.

This paper follows the state-of-the-art steps for carried object detection proposed by D. Damen and D. Hogg in [10] but at the same time approaches each step from a different view point, considering the availability of colour information. For clarity of presentation this paper is divided into a number of sections. Section II aims to familiarize the reader with the system proposed, followed by a detailed step by step presentation of all methods applied, in section III. Subsequently section IV provides a detailed presentation of experimental results and analysis conducted for each step of implemented improvements, followed by a conclusion and suggestions for further research in section V.

II. SYSTEM OVERVIEW

The work of D. Damen and D. Hogg is based on the creation of a temporal template of a moving person and further analysis of its shape to detect the carried object. The concept of temporal template was firstly defined by I. Haritaoglu in [13] as the average of foreground silhouettes

\[ C'(x, y) = \frac{I(x, y) + w^{-1}(x, y)C^{-1}(x, y)}{w^{-1}(x, y) + 1} \] (1)

Where \( I(x, y) \) is the intensity of the foreground pixel, and \( w^{-1}(x, y) \) is the number of times that the specific pixel in \( C(x, y) \) has been classified as foreground. Since we wish to use colour information, we decided to create a colour temporal template. Therefore, the baggage detector takes as input a sequence of colour foreground images which are aligned using image registration to create a colour temporal template (see Fig. 1). Then, the temporal template is matched against exemplar temporal templates of 13 different sizes, 8 viewing directions, and 7 rotations until the best match is found. Subsequently the temporal template is analysed to CIELAB colour space components and the exemplar is subtracted from each of them to reveal the protruding regions which are likely to be carried objects. For further enhancement of the accuracy, D. Damen adopts trained view specific models that map the probable bag location and are used to weight the protruding regions. However, we question the effectiveness of these priors and, for our application we chose a more conservative prior of homogeneous nature to weigh the temporal templates.
It is common sense that the location of the bag and its visibility depends on the position of the viewer in relevance to the position, and moving direction of the pedestrian. This fact brings the need of classifying the temporal template into 8 categories, according to the direction of motion. One way to estimate the Viewing Direction (VD) is to transfer the motion from the image plane (the two-dimensional image surface) to the ground plane (which is at 90 degrees to the picture plane; commonly the ground that objects move on) via the homography transformation as proposed D. Damen in [10]. The problem here is that the homography transformation demands at least 4 corresponding points between the two planes. When the points are provided manually to the system, it becomes impractical while trying to automatically process videos from different cameras or the same pan-tilt-zoom (PTZ) camera. Certainly, automatic methods for homography calculation such as presented in [21] can be used but they are complicated and computationally expensive. In this paper we introduce a novel approach to VD estimation based on template shape features in combination with motion information as in [34].

Fig. 1. A summary of the proposed baggage detection system.

Fig. 2. Temporal templates: (a) is the colour temporal template generated using subpixel image registration, (b) is the corresponding frequency temporal template and (c) is the frequency template generated using ICP.

III. STEP BY STEP PRESENTATION

A. Generation of the temporal template

First step is the generation of the colour temporal template. The foreground silhouettes are segmented by a Gaussian Mixture Model (GMM) background modelling algorithm as proposed by P. Kaewtrakulpong in [17]. Afterwards, a simple connected components tracking algorithm is applied to define the trajectories of the pedestrians. Since all our test videos involve only pedestrians, no human detection algorithm was employed. The extracted coloured foreground permits the utilisation of registration methods based on colour information. The report of R. Szeliski [33] suggests that direct pixel based alignment could be a solution, but is computationally expensive. Hence, Fourier transform is applied to the non-registered images to achieve fast and reliable alignment. Such an algorithm is proposed in [12] as subpixel image registration and its MATLAB source code is freely available for research purposes. This algorithm requires a small pixel sample to perform the registration and is accurate and fast enough to align a long sequence of images. The shoulder area has been selected as the pixel sample, since it remains stable over time unlike the limbs. The output of the algorithm is a translation vector and the error rate. If the error rate is larger than 29% or if the translation vector suggests displacement greater than 18 pixels then the image is not registered to the preceding one to avoid large inconsistencies in the temporal template. Initially, all the silhouettes of a trajectory are centred and cropped to an image of fixed size $h \times w$ and then aligned by two consecutive frames at a time, via subpixel registration. The colour temporal template (CTT) is defined as the average of all silhouettes in each RGB channel using the formula in (1). To differentiate the colour temporal template from the conventional one that is the average of binary silhouettes, as defined in [10], henceforth we will call it a frequency temporal template (FTT). Fig. 2 shows an example comparing subpixel image registration with Iterative Closest Point (ICP) image alignment method employed in [10]. To create the corresponding FTT of the CTT we simply applied the translation vector calculated, to the binary silhouettes.

Having a colour temporal template it is important to decide upon which colour space properties best represent reality. We decided to utilise the CIELAB colour space in [29] because it approximates human vision and therefore has an inherent property of segmentation based on colour. Because of the fact that the $a^*$ and $b^*$ derivatives of the $La^*b^*$ image might have negative
pixel values, the negative values have been transferred to the positive axis and the pixel values have been linearly adjusted. The purpose of creation of these images will be analysed subsequently in this paper.

B. Viewing Direction (VD) estimation

Our primary aim was the complete disengagement of the system from the need for human intervention and the design of a reliable method for direction estimation. Thus, we combine the motion information as in [34] with shoulder shape properties. Fig. 3 reveals that pairs of the 8 exemplars look similar and the shoulder region follows a distinctive pattern. Therefore, we could reduce the classes to five as in Fig. 4 or even to three as follows: class 1 (directions 1 and 5), class 2 (directions 2, 4, 6, and 8), class 3 (directions 3 and 7).

To present the features that we have used for classification we first need to identify some landmark points that would serve the calculation of the features. Initially the frequency template is thresholded with Otsu’s method (histogram based thresholding method proposed by N. Otsu in [23]) to determine the binary image and then the height of the silhouette is defined as the difference of the topmost and bottommost pixels of the binary silhouette. Since the human body is proportionate we take advantage of its properties to localize the head and shoulders. At this point it should be mentioned that although the calculations seem to be based on assumptions that probably do not apply to each and every individual, for the purpose of this method the accuracy of definition of body parts is satisfactory. The only requirement for this technique is that the camera elevation is at a reasonable height. Subsequently we will define all features as variables with mathematical types.

If the height of the silhouette is $H$ and the top of the head is $t$ then the vertical shoulder position (along the $y$ axis) and the point that the image will be cropped is defined as $cut=t+H/4$ and the vertical centre of the head is defined as $VertHeadCent=t+H/17$. The sum of the pixels located at the horizontal line through the vertical path from $y=cut$.

Thus the first feature is defined as

\[
Head _ \_ Shoulder Ratio = \frac{HeadW}{ShouldW}
\]  

(2)

Next we are looking for the horizontal position (along the $x$ axis) of shoulders and head. Thus, the first nonzero pixel at position $x=cut$ is the left shoulder $HorShouldL$ and the last nonzero pixel is the right shoulder $HorShouldR$. Similarly the first non-zero pixel at the vertical position $VertHeadCent$ is the left head side $HeadL$ and the last one the right head side $HeadR$. The centre between two head sides is considered to be the horizontal head center $HorHeadCent$. Thus the next feature is the,

\[
Distance Ratio = \frac{|HorShouldR - HorHeadCent|}{|HorHeadCent - HorShouldL|}
\]  

(3)

![Fig. 3. Exemplar temporal templates of 8 viewing directions.](image)

![Fig. 4. The shoulder areas and their edges that are used to extract information for classification. Here are depicted the exemplars 1 through 5.](image)

For the rest of the features we need the horizontal edges of Sobel derivative of the silhouette as shown in Fig. 4. The features that we extract are the horizontal shoulder and head ranges: $HorShouldRangeL$, $HorShouldRangeR$, and $HorHeadRange$ which is right above the $VertHeadCent$. The reason that we selected these features becomes obvious from Fig. 4 where the visible right and left shoulder width changes in accordance with the viewing direction. In addition, some vertical shoulder features should be taken into account. To calculate the vertical features it is essential to identify the pixels that are at the right and left side of the $HeadR$ and $HeadL$ and simultaneously occur only once through the vertical path from $y=VertHeadCent$ to $y=cut$. Once these pixels are isolated we can compute their vertical mean, range and length which are $meanR$, $meanL$, $rangeR$, $rangeL$, $lengthR$, $lengthL$, respectively. It is also important to examine the slope of the shoulders; thus the found shoulder points are fitted to a line and its slope is denoted as $angleR$ and $angleL$.

The final features are:

\[
Shoulder Range Ratio = \frac{HorShouldRangeR}{HorShouldRangeL}
\]  

(4)
\[ \text{Head Shoulder Range Ratio} = 0.5 \cdot \left( \frac{\text{HorHeadRange}}{\text{HorShouldRangeR}} + \frac{\text{HorHeadRange}}{\text{HorShouldRangeL}} \right) \]  \hspace{1cm} (5)

\[ \text{Shoulder Vertical Range Length Ratio} = \frac{\text{rangeR} / \text{lengthR}}{\text{rangeL} / \text{lengthR}} \]  \hspace{1cm} (6)

\[ \text{Shoulder Height Ratio} = \frac{\text{ShouldW}}{H} \]  \hspace{1cm} (7)

\[ \text{Right Head Shoulder distance} = \text{HeadR} - \text{HorShouldR} \]  \hspace{1cm} (8)

\[ \text{Left Head Shoulder distance} = \text{HorShouldL} - \text{HeadL} \]  \hspace{1cm} (9)

\[ \text{Mean Difference} = \left| \text{meanR} - \text{meanL} \right| \]  \hspace{1cm} (10)

\[ \text{Length Shoulder Difference} = \left| \frac{\text{lengthR}}{\text{ShouldW}} - \frac{\text{lengthL}}{\text{ShouldW}} \right| \]  \hspace{1cm} (11)

\[ \text{Angle Length Difference} = \left\| \frac{\text{angleR} \cdot \text{lengthR}}{\text{lengthR}} - \frac{\text{angleL} \cdot \text{lengthL}}{\text{lengthL}} \right\| \]  \hspace{1cm} (12)

\[ \text{Range Length Difference} = \left| \frac{\text{rangeR}}{\text{lengthR}} - \frac{\text{rangeL}}{\text{lengthL}} \right| \]  \hspace{1cm} (13)

Some of these features, such as Head Shoulder Ratio and Shoulder Height Ratio, are strong and capable of classifying the exemplar template models into 5 classes as in Fig. 4 regardless the size and the in-plane rotation of the exemplar. The effectiveness of these features is demonstrated in Fig. 5 where we can distinguish 5 different classes (represented by 5 dotted clusters where each dot represents an exemplar template). Each cluster corresponds to a different viewing direction where the exemplar templates of different sizes and in-plane rotations share an approximately same feature value. Other features are comparatively weak and act as complementary when the strong ones fail in real life scenarios.

![Fig. 5](image)

Fig. 5. The two figures measure the values of Head Shoulder Ratio and Shoulder Height Ratio respectively as they occur if applied on the exemplars database. Each of the 5 clusters identified consists of 18 exemplars of different sizes and in-plane rotations which are responsible for the small variation of feature values within each class.

SVM classifier was trained to classify into 3 target classes as they were defined earlier in this section. After acquiring the basic information about the pose from the classifier we can identify the VD by combining its pose, and the angle of motion vector (unit vector) from a vertical unit vector (0,-1) on the image plane. The aforementioned motion vector and angles are computed as suggested in [34]. The unit circle in Fig. 6 is marked with the vertical unit vector and angles of high interest, as they define the thresholding areas. Thus, if the angle of the determined motion vector falls in one of these areas then possible directions are the ones indicated within each area in Fig.6. The output of classifier is used to select one of the possible directions in each group. For example if the angle of the motion vector is 85° then the possible directions are 4 and 5. Subsequently, if the output of the classifier is 1 then the final VD will be 5. In case the output of classifier does not match any of the available selections then the most likely direction is selected; in this example 5.
The unit circle with threshold angles on the image plane. The angles separate the circle into segments that suggest the most likely directions within them. The output of SVM classifier indicates which direction should be selected as final.

C. Selection of best matching model and Grow cut segmentation

To find the exemplar model that best matches the under question CTT, we utilise the correlation function. Initially the exemplar model is roughly aligned to the temporal template by head location and its size is approximated by the height of the temporal template as in [34]. To refine the alignment a correlation measure is maximised to confirm the best match.

The correlation function between $h \times w$ matrices A and B is defined as follows:

$$C(A, B) = \frac{\sum_{(x,y)\in[1,1]} (A(x,y) - \bar{A})(B(x,y) - \bar{B})}{\sqrt{\left(\sum_{(x,y)\in[1,1]} (A(x,y) - \bar{A})^2\right)\left(\sum_{(x,y)\in[1,1]} (B(x,y) - \bar{B})^2\right)}}$$

(14)

Since the exemplar models EM are frequency templates (with intensities varying from 0 to 1) the CTT has to be adjusted accordingly. Therefore, it is converted to grayscale, inverted, and normalized to the range [0,1]. The similarity measure function between inverse temporal template $PT$ and the exemplar model EM to be maximised is defined as

$$S(PT, EM) = 0.8C(PT_{tg}, EM) + 0.2C(PT_x, EM)$$

(15)

where

$$PT_{tg}(x,y) = PT(x,y)g(y)$$

(16)

$$PT_x(x,y) = \begin{cases} PT(x,y)g'(y), & \text{if } EM(x,y) > 0 \\ 0, & \text{if } EM(x,y) = 0 \end{cases}$$

(17)

where, $g(y)=[4.40;-0.01:2.01]$ and $g'(y)=[3.20;-0.005:2.005]$

$EM_{best}$ is selected as the best matching transformed exemplar that maximizes $S(PT, EM)$

$$\arg \max_{EM} S(PT, EM) = \{EM_{best}\}$$

(18)

For further enhancement of accuracy the selected exemplar template is dilated by square structuring element $SE_{ij}$, $i, j \in \{1, 2\}$.

$$EM_{best} = \begin{cases} EM_{best} \oplus SE, & \text{if (7) for Diff}(x,y) \text{ is } > 0.1 \\ EM_{best}, & \text{otherwise} \end{cases}$$

(19)
where, \( \text{Diff}(x, y) = \left| \text{PT}(x, y) - \text{EM}_{best}(x, y) \right| \)

The threshold for the implementation of dilation is selected after testing through our datasets.

Sometimes the shape of the best matching exemplar is not enough to cover the torso area. Hence, GrowCut algorithm [35] is employed to define the torso shape. In more detail, the Grow cut algorithm takes as input two images: the seeds and the image on which the algorithm is applied. The seeds are the labels for the foreground, background and the unknown regions. We will call the seed image as mask \( M \) and it will be labelled as follows:

\[
M(x, y) = \begin{cases} 
1, & \text{if foreground} \\
-1, & \text{if background} \\
0, & \text{if uncertain}
\end{cases} \quad (20)
\]

Foreground is a predefined pixel selection as shown in Fig. 7 which is transformed according to transformations applied to the best matching exemplar to match the size and rotation of the temporal template. Background is all the pixel values greater than 200 on the CTT, and unknowns are all the rest. The \( a^* \) component of \( L_{a*b*} \) image representation is selected as the second input argument. Thus the result of segmentation are the labels indicating the foreground region; here the torso (see Fig. 8).

\[
\text{labels} = \text{GrowCut}(a^*, M) \quad (21)
\]

As mentioned above, the torso segmented via GrowCut is used to redefine the exemplar itself. So, if the blurred GrowCut foreground is \( B \) (Fig. 9 (b)) then the final best matched exemplar template \( \text{EM}_f \) is defined in (22) (see Fig. 9 (d)).

\[
\text{EM}_f(x, y) = \min(1, B(x, y) + \text{EM}_{seg}(x, y)) \quad (22)
\]

Difference image \( v(x, y) \) calculated in (23) reveals the protruding regions that undergo further analysis for carried objects extraction (Fig. 9 (f)). The corresponding differences for \( a^* \) and \( b^* \) components of the \( L_{a*b*} \) temporal template are given by (24) and (25).

\[
v(x, y) = \max(0, \text{PT}(x, y) - \text{EM}_f(x, y)) \quad (23)
\]

\[
v_a(x, y) = \max(0, \text{PT}(x, y) - a^*(x, y)) \quad (24)
\]

\[
v_b(x, y) = \max(0, \text{PT}(x, y) - b^*(x, y)) \quad (25)
\]

Fig. 7. Direction specific mask images used as input seeds to GrowCut algorithm.

Fig. 8. GrowCut segmentation: (a) is the temporal template, (b) is the selected best matching exemplar, (c) is the transformed (resized and translated) mask, (d) is the GrowCut segmentation results (e) is the difference image (f) is the labelled bags.

Fig. 9. Redefinition of exemplar with the segmented torso. (a) is the temporal template, (b) is the blurred GrowCut result (c) is the best matching exemplar (d) the combination of exemplar with the segmented torso. (e) is the inverse grayscale temporal template and (f) the difference image \( v(x, y) \).
D. Segmentation by energy function minimization

The segmentation of bags is handled as a labelling problem by D. Damen in [10] where the segmentation of carried objects is achieved via energy minimisation using the GraphCuts algorithm [4]. Therefore the pixels that belong to carried objects should be labelled as foreground and limbs as background. Assuming that the label assigned to a certain pixel \( p_i(x, y) \) depends only on the labels assigned to its neighbouring pixels, the difference image \( v(x, y) \) can be considered as a first-order Markov Random Field (MRF). Since we assume a 4-neighbourhood system on a 2D lattice and the label set is \( L = \{0, 1\} \), the Gibbs energy function follows a special case of the Ising model [20]. If \( P \) is an image (lattice) \( h \times w \) and \( N \subset P \) is the neighbourhood system for \( P \) defined as \( N = \{N_p \mid \forall p \in P\} \), then according to Bays’ theorem to achieve segmentation a maximum a posteriori (MAP) solution should be given (i.e. maximizing the posterior probability). This is equivalent to minimizing the following energy function

\[
E(f) = \sum_{p \in P} D_p(f_p) + \sum_{p \in \partial N} V_{p,q}(f_p, f_q)
\]

(26)

Where \( D_p \) is the data cost function, and \( V_{p,q} \) is the smoothness cost or click potential function that defines the interaction between the neighbouring pixels. Supposed \( f = \{f_1, \ldots, f_{h \times w}\} \) is a set of labelling configurations defined on the lattice \( P \) such that \( f_p = \hat{f}(p), \forall p \in P \), then \( f_p \) assigns a label \( L \) to the pixel \( p \in P \), [20], [18]. The data cost function \( D_p \) can be an arbitrary probability density function (p.d.f) that measures the cost of assigning the configuration \( f_p \) to a pixel. Therefore, we will search for the p.d.f that best describes the given data. In our case the data cost function is defined as follows.

\[
D_p(f_p) = -\ln \left( p_1(v(x, y) \mid f_p) + p_2(v(x, y) \mid f_p) \right)
\]

(27)

And the smoothness cost is

\[
V_{p,q}(f_p, f_q) = \begin{cases} s & \text{if } f_p \neq f_q \\ 0 & \text{if } f_p = f_q \end{cases}
\]

(28)

The class conditional distributions are

\[
p_1(v(x, y) \mid f_p) = \begin{cases} \frac{1}{\kappa} M(x, y) * \hat{W}(y) & \text{if } f_p = 1 \\ \kappa \ln N(x, y) & \text{if } f_p = 0 \end{cases}
\]

(29)

\[
p_2(v(x, y) \mid f_p) = \begin{cases} (\lambda v_u(x, y) + \lambda v_b(x, y)) * \hat{W}^{-1}(y) & \text{if } f_p = 1 \\ -(\lambda v_u(x, y) + \lambda v_b(x, y)) * \hat{W}^{-1}(y) & \text{if } f_p = 0 \end{cases}
\]

(30)

The p.d.fs were selected as the mixture \( M \) of normal distributions and log-normal (\( \ln N \)) distribution, multiplied by constants \( 1/\kappa \) and \( \kappa \) respectively. Finally, \( \hat{W} \) is a Gaussian weight function and its purpose will be clarified later on.

The coefficient \( \lambda \) is used to weigh the \( v_u(x, y) \) and \( v_b(x, y) \) difference images.

1) Utilization of La*b* colour space derivatives a* and b*

Due to the fact that the difference images \( v_u(x, y) \) and \( v_b(x, y) \) contain noise that occurs because of the significant colour information present at the outline of the silhouette, the connected components that are unlikely to represent a bag have been removed by considering their aspect ratio, their size relative to the silhouette and the pixel intensity. Thus, the connected components that have been removed could be characterised as unusually long, small or lacking colour.

In detail, the separation between object and non-object areas is executed in two phases: in the first phase the thresholds are applied directly to the binary \( v_u(x, y) \) and \( v_b(x, y) \) images and in the second phase the thresholds are applied to the binary \( v_u(x, y) \) and \( v_b(x, y) \) images after morphological erosion. The values that are thresholded for each object, in both phases, are: the ratio of the major and minor axes lengths, the area, the mean value of pixel intensities and an additional aspect ratio feature for the second phase. Subsequently, the results of both phases are combined into one image and its equivalent grayscale image is obtained from the original \( v_u(x, y) \) and \( v_b(x, y) \) images.
2) Definition of the weight function $\hat{W}$

As it is expected some of the areas with high probability do not always belong to a carried object, especially the ones that are located around the head or well under the feet. For example, the face colour contains high intensity values, due to which the head region is extracted from the $a^*$ and $b^*$ images as potential carried object on a number of occasions. Therefore there is a need for a function that weighs the difference images $v(x,y)$, $v_d(x,y)$, and $v_h(x,y)$ to reduce the probability in the unlikely areas. The weight function is constructed as a vector that will be multiplied with each column of the difference image.

A linear function does not assist our aim since the transition across the curve is very steep. On the other hand the Gaussian function offers a smooth and slow transition and is evaluated as the most suitable due to its bell shape (31).

$$f(x) = a \exp\left(-\frac{(x-h)^2}{2\sigma^2}\right)$$  \hspace{1cm} (31)

Thus, for a given vector $x=[1, 2, ..., h]$, where $h$ is the height of the temporal template image, we attempt to construct a vector $W$ of size $h \times 1$ that contains all the weights in the following way:

Let us assume two different vectors that obey two different Gaussian functions,

$$(W_{a^*})_{i=1}^h = [(d_i)_{i=1}^h \ (c_i)_{i=1}^h]$$  \hspace{1cm} (32)

$$(W_{b^*})_{i=1}^h = [(c_i)_{i=1}^h \ (d_i)_{i=1}^h]$$

where $(c_i)_{i=1}^h = (c_i)_{i=1}^h = 0.5$ and

$$(d_i)_{i=1}^h = 0.5\varepsilon', \hspace{1cm} \text{where} \hspace{1cm} t = -\frac{((x_i)_{i=1}^h - h / 2)^2}{2(100(S_a - 0.2))^2}$$

$$(d_i)_{i=1}^h = 0.5\varepsilon', \hspace{1cm} \text{where} \hspace{1cm} t = -\frac{((x_i)_{i=1}^h - h / 2)^2}{2(100(S_a + 0.2))^2}$$

Because the size of the temporal templates differs it would be unfair to use the same weight vector for different sizes. Hence, the weight vector is adapted according to the size $S_a = \{0.2, 0.25, 0.3, 0.35, ... 0.8\}$ of the best matched exemplar so that the higher probabilities are concentrated around the silhouette’s torso. The updated weight vector $\hat{W}$ is,

$$(\hat{W}_{a^*})_{i=1}^{10} = (W_{a^*})_{k \rightarrow 0.5S_a}$$

$$(\hat{W}_{b^*})_{i=1}^{10} = (W_{b^*})_{k \rightarrow 0.5S_a}$$  \hspace{1cm} (33)

To clarify the concept Fig. 10 demonstrates how exactly $\hat{W}$ looks like and the results of its multiplication with difference images.

3) Definition of the probability function $\mathcal{M}(x,y)$

As it was proposed by D. Damen the pixel values under a threshold constitute to noise and all the rest are likely to belong to a carried object. This threshold is decided using the following equation $\text{thresh} = \min\left(0.38, \max\left(\{v(x,y)\}/1.72\right)\right)$. As an example, the distribution of pixels greater than the threshold is presented in Fig. 11. Since the pixel values on our grayscale temporal template vary with the colour of the clothing and the carried object, the conditional distribution cannot be approximated by a single distribution and cannot be the same for all the cases. Thus, a non-parametric density estimation technique is used to obtain an intuition of how the pixel values are distributed and to determine the parameters of the Gaussian distribution or the mixture of Gaussian distributions that would form the likelihood function.
The averaged shift histogram is a common non-parametric density estimation technique which defines the probability of a pixel $p$ having a value $x_i$ out of a sample of values $x_1, x_2, \ldots, x_n$ and given the kernel function $K(t)$ according to [30] as follows.

$$ P_r(x) = \frac{1}{nh} \sum_{i=1}^{n} K \left( \frac{x - x_i}{h} \right) $$

where $n$ is the sample size, and $h$ is the smoothing factor, also known as bandwidth. Built in MATLAB routine “ksdensity” was used to perform density estimation having normal distribution kernel function $K(t)$ and $h$ computed as suggested by Silverman in [32]. Hence

$$ h = \hat{\sigma}C_v(k)n^{-1/(2v+1)} $$

where $\hat{\sigma}$ is the sample standard deviation, $v$ is the order of the kernel and $C_v(k)$ is the constant for the chosen distribution, which for a Gaussian kernel of size $v=2$ is equal to 1.06. Therefore, we can simply write $h = 1.06\sigma n^{-1/v}$. The above density estimation was applied to the difference image $v(x,y)$. The main purpose of this operation is to smooth out the intensity histogram to obtain the mean values of the mixture of Gaussian distributions. For example given the curve in Fig. 11 we separate our sample into two groups as indicated by local minima and each subsample has its mean at local maxima. From our experiments we have found that in the case of a true detection, the standard deviations of the distributions are proportionate to the computed mean values. Hence, $s$ is found to be

$$ s = \left( \frac{\bar{x} - \text{thresh}}{0.01 - b} \right) + b $$

where $b=\text{thresh}-0.09$, $s$ belongs to the interval $(0.01, b)$ and $\bar{x}$ to $(\text{thresh}, 1)$. The above function establishes a correspondence between the mean and the standard deviation. The likelihood mixture $\mathcal{M}$ of Gaussians is computed in such a way that the sum of their coefficients $k_i$ approximates 1. Therefore, if the number of found distributions is $G_n$, then

$$ \mathcal{M} = \sum_{i=1}^{G_n} k_i \mathcal{N}(\bar{x}_i, s_i) $$

The coefficient values $k_i$ are decided to be analogical to the highest probability for each mean value; this is analogical to the local maxima of the non-parametric density curve. Thus,

$$ k_i = \frac{\max_j}{\sum_{j=1}^{G_n} \max_j}, i = \{1 \ldots G_n\} $$

where, $\max_j$ are the local maxima.

4) Definition of the probability function $\ln \mathcal{N}(x,y)$

By observing the first histogram in Fig. 12, which represents the distribution of the pixel values smaller than the defined threshold ‘thresh’, we conclude that they follow a log-normal distribution. To find the parameters of the log-normal distribution it is required to plot the log values of the pixels [27]. Since the second histogram in Fig. 12 resembles a normal distribution, the
mean value for the log-normal distribution is approximately $\mu = -0.7$ and the standard deviation is $\sigma = 2.2$ as obtained from the normal distribution histogram.

Fig. 11. An example showing non parametric density curve with two peaks/two means (left). Proposed technique where standard deviations are adjusted proportionally to the means and the resulting curve (right). The corresponding templates are on either side of the curves.

Fig. 12. The noise distribution for pixel values <5.8 (left) and the distribution of their log values (right).

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In order to critically evaluate the performance of the improved system a significant number of detailed experiments were conducted. The data used for the experiments were obtained from the PETS 2006 dataset and further videos recorded by the authors using a hand-held and CCTV cameras to cover specific test scenarios that were not present in the PETS dataset. Since the system is not capable of processing occluded silhouettes, these have been removed manually. The experiments have been conducted on a computer with a 2.53 GHz processor and 4.00GB RAM memory. Each person’s trajectory has been split in such a way that it records 2 seconds of movement (approximately 2 walking cycles) as suggested by D. Damen. Therefore, hereafter we will refer to each part of a trajectory as a separate individual and consequently the sample size will increase accordingly. The final sample size will be 239 (or 179 not split) individuals for the PETS dataset, 95 (or 39 not split) for the in-house videos obtained by the hand-held camera, and 94 (or 45 not split) for videos obtained via CCTV cameras. The implementation was carried out in MATLAB and constitutes a major revision of the initial implementation by D. Damen and Hogg. The following subsections present the experimental results for each step of the procedures described in this paper.

A. Temporal template generation and La*b* colour space exploitation

As it was explained in the relevant section, the temporal template is generated by averaging the extracted foreground silhouettes. To that end, all the images through the frames have to be aligned. Here, we aim to prove that the proposed image registration method offers a considerably better outcome than ICP alignment. The figures that follow, compare the ICP and Subpixel image registration methods (see Fig. 13 and 14). Undoubtedly, the Subpixel registration has outperformed the ICP in terms of accuracy and time.

Another division of our system that needs to be justified is the GrowCut exploitation and La*b* colour space selection. The main reason for Grow cut application is the definition of human torso. We attempt to take advantage of the fact that the colour of clothes is uniform over the entire torso region and the bag colour differs from that of the clothes. In a sense, we utilise the high contrast attribute of $a^*$ and $b^*$ components of the La*b* colour space. Fig. 15 illustrates the results of applying GrowCut segmentation on the $a^*$ component. Apparently, the segmentation is not successful on white colour clothes due to the lack of colour information (Fig. 15 (g)).

From the viewpoint of definition of bags the La*b* colour space offers significant improvement to the system. Whereby, not only it enables the detection of otherwise undetected objects, but it also contributes to the accurate acquisition of bags’ shape by reducing the noise caused by shadow. Fig. 16 shows some of the successfully and unsuccessfully isolated bags. By observing the presented examples it is possible to understand which colours are encoded by each of the $a^*$ and $b^*$ components and the reason that some bags are not segmented; i.e. mainly the weak colour representation of the bag including considerable amount of black and white colour.
Fig. 13. Comparison of image alignment techniques. The first and second rows illustrate the frequency templates created by using subpixel image registration and ICP respectively.

Fig. 14. Processing time comparison of image alignment techniques.

B. Direction estimation evaluation

The SVM classifier has been trained to obtain the pose of the pedestrians. We chose the LIBSVM library as a tool to implement the classification [7]. As matter of fact, the classifier can recognize between three different poses. The motion vector of the pedestrian is used to refine the results and classify the object to one of the 8 categories. Since the performance of the classifier cannot be evaluated objectively over the whole dataset, 8 random images have been selected from each direction group to test the classifier; thus the testing set consists of 8*5=40 samples.

Fig. 15. Human torso definition: The first row depicts the CTT, while the second row their a* component of the La*b* colour space. The result of GrowCut algorithm as a mask over the temporal template is shown in the last row. The algorithm fails while expanding over colourless areas like in (g).

Fig. 16. Segmentation of the carried objects by means of a* and b* derivatives of the CIELAB colour space.

Initially, we have selected 12 images from all available datasets for each of the 5 groups shown in Fig. 4 and labeled them with 3 different labels, depending on their posture. Hence the training set consists of 12*5=60 real life samples. As the set of training images is not large enough for training and not all the features are rotation and scale invariant, the 60 temporal templates have been flipped vertically and rescaled to a range of [0.5:0.1:1]; in total obtaining 60*6*2=720 samples. A further set of 90 exemplar templates of selected sizes and rotations have been added as a supplement to the training set.
The classifier has been optimized by performing 6-fold cross-validation and “grid-search” as suggested by C. Hsu et al. in [15] to select the kernel function and the training parameters. The set was separated in a way so that the rescaled versions of each image were at the same group; i.e. testing or training. The set of 90 exemplars was included in each training fold, since it always gives accuracy of 100%. Among linear, polynomial and Radial Basis kernel functions, the latter one was selected since it gives the highest average accuracy of 85.6 for the training parameters \(C=0.46\) and \(\gamma=0.004\). The final classification model was tested over the sample of 40 silhouettes and the accuracy achieved was 80%.

To compare the direction estimation method used by D. Damen with the proposed one, the 3rd camera view videos have been selected from the PETS dataset. The rest of the datasets in Table 1 simply demonstrate the capability of the system to identify other viewing directions as well (CCTV videos: mostly directions 3 and 7, videos captured using handheld cameras: mostly directions 1 and 5). Although the accuracy values are high, they do not approximate to 100%. This is due to the deformation of shoulder shape, either as a result of poor temporal template generation or objects carried on the shoulders.

### C. Overall performance

After applying all proposed improvements it is important to examine their collective contribution to the overall performance of the system. Due to the fact that the system cannot deal with solely occluded objects, they were not annotated as ground truth. Since the detection is enhanced by various factors it is vital to check the degree of enhancement that they offer at each stage. Numerous experiments were conducted on all our datasets changing the different components of the system.

It is obvious that the appearance of coefficients at different stages is quite frequent and their selection was not a trivial task. The performance of the system has been thoroughly examined for the different values of the coefficients over the PETS dataset and the best ones were tested on the other two datasets for the selection of the final ones. Therefore the selected coefficients in (29) and (30) are \(K=0.8\), \(\lambda=6\).

To begin with, the significance of \(La*b*\) colour information to the detection of bags should be examined. It is also important to incorporate the energy function as defined by D. Damen into our system and see how exactly the two different energy functions influence the final results when applied on the same data. Since the effectiveness of the trained models is questionable, they are replaced with the gradient weight \(\nabla^2\) and the \(La*b*\) images in D. Damen’s energy function. Afterwards, the energy function that produces the best results is compared with the original method proposed by D. Damen.

### Table 1

<table>
<thead>
<tr>
<th>Proposed Method</th>
<th>PETS 2006</th>
<th>Hand-held camera videos</th>
<th>CCTV videos</th>
</tr>
</thead>
<tbody>
<tr>
<td>D. Damen’s Baggage detector</td>
<td>0.90</td>
<td>0.89</td>
<td>0.81</td>
</tr>
<tr>
<td>0.83</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table 2

<table>
<thead>
<tr>
<th>A. Energy Function as defined by D. Damen. (v_D(f_{fpq}) = 2.8)</th>
<th>PETS dataset 3rd camera view</th>
<th>Hand-held camera videos</th>
<th>CCTV videos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed Method &amp; D. Damen and substituting the trained bags model with gradient weight and (a<em>b</em>) images. (v_D(f_{fpq}) = 2.8)</td>
<td>0.57</td>
<td>0.60</td>
<td>0.68</td>
</tr>
<tr>
<td>CCTV</td>
<td>0.55</td>
<td>0.83</td>
<td>0.58</td>
</tr>
<tr>
<td>Total</td>
<td>0.58</td>
<td>0.68</td>
<td>0.63</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>B. Energy Function as proposed in this paper. (v_D(f_{fpq}) = 3)</th>
<th>PETS dataset 3rd camera view</th>
<th>Hand-held camera videos</th>
<th>CCTV videos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed Method &amp; D. Damen and substituting the trained bags model with gradient weight and (a<em>b</em>) images. (v_D(f_{fpq}) = 2.8)</td>
<td>0.61</td>
<td>0.65</td>
<td>0.66</td>
</tr>
<tr>
<td>CCTV</td>
<td>0.65</td>
<td>0.91</td>
<td>0.66</td>
</tr>
<tr>
<td>Total</td>
<td>0.61</td>
<td>0.74</td>
<td>0.62</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>C. Energy Function as proposed in this paper. (v_D(f_{fpq}) = 3)</th>
<th>PETS dataset 3rd camera view</th>
<th>Hand-held camera videos</th>
<th>CCTV videos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed Method &amp; D. Damen and substituting the trained bags model with gradient weight and (a<em>b</em>) images. (v_D(f_{fpq}) = 2.8)</td>
<td>0.54</td>
<td>0.72</td>
<td>0.46</td>
</tr>
<tr>
<td>CCTV</td>
<td>0.48</td>
<td>0.95</td>
<td>0.44</td>
</tr>
<tr>
<td>Total</td>
<td>0.51</td>
<td>0.80</td>
<td>0.40</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>D. Energy Function as proposed in this paper without the (a<em>b</em>) enhancement. (v_D(f_{fpq}) = 3)</th>
<th>PETS dataset 3rd camera view</th>
<th>Hand-held camera videos</th>
<th>CCTV videos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed Method &amp; D. Damen and substituting the trained bags model with gradient weight and (a<em>b</em>) images. (v_D(f_{fpq}) = 2.8)</td>
<td>0.53</td>
<td>0.57</td>
<td>0.66</td>
</tr>
<tr>
<td>CCTV</td>
<td>0.54</td>
<td>0.79</td>
<td>0.59</td>
</tr>
<tr>
<td>Total</td>
<td>0.41</td>
<td>0.44</td>
<td>0.25</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>E. Energy Function as proposed in this paper with D. Damens best model match selection (i.e. template-exemplar alignment). (v_D(f_{fpq}) = 3)</th>
<th>PETS dataset 3rd camera view</th>
<th>Hand-held camera videos</th>
<th>CCTV videos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed Method &amp; D. Damen and substituting the trained bags model with gradient weight and (a<em>b</em>) images. (v_D(f_{fpq}) = 2.8)</td>
<td>0.53</td>
<td>0.57</td>
<td>0.66</td>
</tr>
<tr>
<td>CCTV</td>
<td>0.54</td>
<td>0.79</td>
<td>0.59</td>
</tr>
<tr>
<td>Total</td>
<td>0.41</td>
<td>0.44</td>
<td>0.25</td>
</tr>
</tbody>
</table>
Table 2 shows all the results for the 3 datasets and 4 different energy functions. The total is calculated by combining the results from all datasets, not by averaging the percentages obtained. The cases selected for examination are: A. the energy function as defined by D. Damen in [15] applied on the FTT and combined with the trained bags models, B. the same D. Damen’s energy function but with gradient weight $\tilde{H}$ and $\mathbf{a}^* \mathbf{b}^*$ enhancement instead of the trained bags models, C. the energy function as defined in this paper applied on the CTT, D. the same energy function but without the $\mathbf{a}^* \mathbf{b}^*$ information, and finally E. the proposed energy function but with D.Damen’s best model match search. By examining the table it can be inferred that the $\mathbf{a}^* \mathbf{b}^*$ images offer a substantial enhancement to our system and to D. Damen’s energy function as well. If not employed as in case D we give up approximately 20% of true positive detections for 10% of true negative detections and as a consequence the total accuracy level falls to 51%.

The low precision and specificity values in case A of the Table 2 show that the D. Damen’s energy function suffers from a significant number of false positives. Our aim was to retain the recall values but at the same time to increase the precision and the definition of the shape of the segmented bags. Concerning the alignment of the temporal template with the best matching exemplar the cases C and E should be considered. It is obvious that D. Damen’s method does not serve the requirements of our system and therefore it has been modified.

In general the specificity numbers are low which shows the weakness of the system in recognizing the negative cases. However the accuracy and precision levels have been increased. We encourage the readers to further observe Table 2 and make their own conclusions.

Case C, which is proposed in this paper, provides the best results; therefore, it is the one to be compared to D.Damen’s original system. The results are presented in Table 3.

![Fig. 17. Precision-Recall curves for the final improved system in case C of Table 2 for the 3 different datasets.](image1)

![Fig. 18. Receiver operating characteristic curves for the final improved system in C of Table 2 and the primary one as proposed by D. Damen.](image2)
The ground truth box for the position of the bags on the temporal templates was obtained manually. Detection is considered as successful only if the overlap between the ground truth bounding box and the detected one is higher than 20%. In case the overlap is between 0% and 20% then, if the bounding box obtained is inside that of the ground truth then it is labeled as false negative, else if the ground truth box is inside the bounding box obtained then it is labeled as a false positive. Any other case would suggest that the bounding box obtained and the ground truth box is not related to each other and therefore the detection is labeled as a false positive and a false negative, respectively.

The Precision-Recall (PR) curves in Fig. 17 are derived from each of the three available datasets while the ones in Fig. 18 and 19 are only derived from the PETS dataset. They are the result of linear interpolation of recall points corresponding to maximum precision. The receiver operating characteristic (ROC) curves in Fig. 18 are constructed as complementary to the PR curves. In Fig. 17 the results are very encouraging for the in-house videos captured via the handheld cameras; this is because the camera angle and distance is optimized for the purpose of the application. The curves in Fig. 18-19 compare the proposed method with that of D. Damen’s, for the PETS dataset. The ROC curves show that the false positive rate has been eliminated and the true positive rate has been increased.

The last metric introduced is the Average Precision (AP), which summarises the shape of the precision/recall curve, and is defined as the mean precision at a set of eleven equally spaced recall levels \([0, 0.1, ..., 1]\) [28]:

\[
AP = \frac{1}{11} \sum_{r \in [0, 0.1, ..., 1]} p_{\text{meas}}(r)
\]  

(39)

The precision at each recall level is interpolated by taking the maximum precision measured for all recalls greater than \(r\)

\[
p_{\text{meas}}(r) = \max_{\tilde{r} > r} p(\tilde{r})
\]

(40)

where \(p(\tilde{r})\) is the measured precision at recall \(\tilde{r}\). The calculated AP for the improved version is 0.64 while for the D. Damen’s system it is 0.49. These results show the significance of the improvement of the system.

V. CONCLUSION

In this paper we have proposed a number of innovative methods that incrementally improve the overall performance of the baggage detection system originally proposed by D. Damen.

Our first goal was to compose a robust direction estimation algorithm. For this reason we have exploited the pedestrian’s shoulder shape features and their displacement over the image plane. However, not all available motion information has been fully utilized. For example the duration of motion on the scene and the silhouette size variation during motion could indicate the direction of motion. Hence, there where objects’ size remains approximately the same as they move, tend to cross the scene horizontally, while the ones that are getting smaller in size tend to walk vertically. These are perspective related properties that could be employed to enhance our algorithm.
The next aim is to enhance the clothes and torso detection of the person. Therefore, the GrowCut algorithm used can be combined with colour clustering techniques to improve the accuracy of torso detection. In such case the torso of the best matched template should be substituted by the one obtained. Another important aspect was the segmentation of bags using the $a^*$ and $b^*$ components of the $La^*b^*$ colour space. Since the $La^*b^*$ colour space values change with illumination, good illumination conditions are in favour of the proposed method. Towards this direction, the system has been tested under 3 different illumination conditions, where 2 of them were outdoors. The achieved results were encouraging.

To sum up the system has been improved not only in terms of accuracy and precision but also in accurate bag shape segmentation. However, the method still suffers from some inaccurate alignments of the exemplar with the temporal template and some inaccurate direction estimations. Other major challenges that need research further attention are loose clothes and partially occluded or small objects.

REFERENCES


