Enhancing pedestrian detection using optical flow for surveillance

Redwan A.K. Noaman*, Mohd Alauddin Mohd Ali and Nasharuddin Zainal

Department of Electrical, Electronic and Systems Engineering, Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia, 43600 Bandar Baru Bangi, Selangor, Malaysia
Email: redwan_eng@yahoo.com
Email: mama@eng.ukm.my
Email: nash@eng.ukm.my
*Corresponding author

Abstract: Optical flow can be used to segment a moving object from its backgrounds and track it. In this paper, an Enhanced Lucas-Kanade optical flow technique was used to improve human detection in terms of speed and accuracy. We combined object segmentation output with a human detector using an optical flow algorithm. The proposed technique used the optical flow to find the area of interest to complete object segmentation and use those results as an input for the human detector. This technique has been developed to be used in surveillance systems. Our experiments indicated that the proposed method was 37% faster and 118% more accurate than the standard Felzenszwalb (PFF) detector.

Keywords: optical flow; Luckas-Kanade; Gaussian filter; human detection; PFF detector.


Biographical notes: Redwan A.K. Noaman received his BEng in Computer and Control Engineering from the University of Sana’a (Yemen), in 2003. In 2008, he joined as a researcher in the computer and network security group, UKM. He received his MEng in Computer and Communication Engineering from Universiti Kebangsaan Malaysia (UKM), Malaysia, in 2008. Currently, he is working toward his PhD in the Department of Electrical, Electronic, and Systems Engineering, UKM. His current research interests are mainly focused on image processing and computer vision with emphasis on human detection and tracking for surveillance.

Mohd Alauddin Mohd Ali is a Professor at the Department of Electrical Electronic and Systems Engineering, Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia (UKM), Bangi, Malaysia. He was also the Head of the department and Deputy Dean of the faculty for the periods 1995–1997 and 1997–2002 respectively. He also served as the Director of the Institute of Space Science, UKM for the period 2009–2013. He received
his BEng (Electrical), BSc (Mathematics) and MEngSc (Electrical) degrees from the University of Tasmania, Hobart, Australia in 1978, 1979 and 1984 respectively. His PhD degree was from the University of Nottingham, Nottingham, England in 1994. He worked in the area of microelectronics for his Masters degree and medical electronics for his PhD. His research interests include biomedical signal processing, instrumentation, integrated circuit design and testability, and ionospheric studies.

Nasharuddin Zainal received his BEng in Computer Science from Tokyo Institute of Technology (Japan), in 1998, MEng in Communication and Computer Engineering from Universiti Kebangsaan Malaysia (UKM), Malaysia, in 2003, and DrEng degree from Tokyo Institute of Technology (Japan) in 2010. He is currently a Lecturer at the Department of Electrical, Electronic and Systems Engineering, Universiti Kebangsaan Malaysia, Bangi, Malaysia. His main research interests include computer engineering, image and video processing and pattern recognition.

This paper is a revised and expanded version of a paper entitled ‘Enhancing human detection using optical-flow’ presented at International Conference of Recent Trends in Information and Communication Technologies, Universiti Technologi Malaysia, Johor Malaysia, 13 September 2014.

1 Introduction

Detecting human figures is currently an active research topic in computer vision. It is a key enabler for applications in robotics, surveillance, and intelligent transport systems (ITS). It is used for tracking, and recognising people. As a result, automated systems that can estimate and track moving objects have received a lot of attention from Industry and academia for their potential surveillance and engineering applications including video surveillance, content-based image retrieval, and gait recognition (Benenson et al., 2012; Tafazzoli and Safabakhsh, 2006; Yao et al., 2014; Zhang et al., 2014).

Detecting people in images is quite challenging because of their intra-class variability, the diversity of the backgrounds, and the conditions under which the images were acquired. Variations in the images such as clothing, lighting and shape morphing (Dalal and Triggs, 2005) present additional challenges. Even the detection of non-occluded, stationary human figures has been the subject of a number of studies. Several studies have focused on classification systems to improve human detector methods, such as building classifiers that provide an image window that can be used to decide if there is a human figure in the window. Most successful classification processes for human detection learn-from-example, such as in, paradigm (Enzweiler and Gavrila, 2009; Geronimo et al., 2010). Dalal, and Triggs (2005) a proposed holistic classifier that relied on histograms of oriented gradients (HOG) and linear support vector machines (SVM). Paradigm remains as a competitive baseline method for comparing new human classifiers (Dollar et al., 2009; Enzweiler and Gavrila, 2009).

Human detection and tracking in surveillance applications requires the segmentation of objects where the regions of interest (ROIs) are separated from the background. In a scene, a system will typically divide a scene into the foreground and background. Only the foreground contains the items of interest and the background remains relatively unchanged over time. To achieve this separation of foreground and background,
Enhancing pedestrian detection using optical flow for surveillance

Optical flow is an approximation of local image motion based upon local derivatives in a given sequence of images. It specifies how much each image pixel moves between adjacent images. The underlying aim for an optical flow calculation is the conservation of pixel intensity. Optical flow meets this goal by assuming that the intensity or colour of an object has not changed significantly between two frames (Horn and Schunck, 1981).

Human detection and tracking techniques (Enzweiler and Gavrila, 2009; Geronimo et al., 2010) that use optical flow, perform detection and tracking by averaging the flow for the located object in the first frame and searching for a region that has similar flow vectors in the second frame. Recently, a mix of gradient-based methods (Enzweiler and Gavrila, 2009; Geronimo et al., 2010) and block matching-based methods (Dollar et al., 2009; Enzweiler and Gavrila, 2009) have been used to develop human detection and tracking systems. Gradient-based methods analyse the change in intensity and gradient (using partial spatial and temporal derivatives) to determine the optical flow. Block matching-based methods rely on determining the correspondence between two images. This typically involves matching ‘blocks’ of one image to ‘blocks’ of the other image to determine how far that region has moved. Both methods perform best when determining flow at or around clearly defined features, and make assumptions such as constant luminance for a given region across multiple frames. In other words, both methods assume that as an object moves, its appearance does not change due to lighting.

On the other hand, common object and human detection techniques analyse the frame as a whole and use a detection model for every pixel whether the pixel is in motion or not. This leads to large amounts of unnecessary processing and results that may detect unwanted objects. The Lucas-Kanade method is widely used as a differential method for optical flow estimation for computer vision. However, the drawbacks of this method include errors regarding the boundaries of moving object where many unwanted vectors appear due to lighting and camera noise (Turaga et al., 2008). Current research in this area is interested in low level (detection), intermediate level (tracking) and high level (behavioural analysis) (Geronimo et al., 2010). This paper focused on the low level stage of human detection.

All the methods mentioned above require large amounts of computational processing and complex models. In this paper, we enhanced the Lucas-Kanade optical flow in order to improve human detection results using a pre-processing stage before initiating a state-of-the-art detector. Instead of analysing the whole frame, the enhanced Lucas-Kanade technique was used to obtain a region of interest (ROI). ROI is used as the input for the human detector. A Pedro F. Felzenszwalb detector (PFF detector) (Felzenszwalb et al., 2010) was used to detect human in images using a combination of deformable part models (Turaga et al., 2008). This enhancement reduced processing time and increased the accuracy of the detector.

2 Methods

In this paper, we enhanced the Lucas-Kanade optical flow technique and used it to improve the speed and accuracy of the human detector. As shown in Figure 1, every couple of frames were processed using an enhanced optical flow technique to obtain motion vectors (map of vectors). Next, a Gaussian special filter was used to reduce unwanted motion vectors to avoid incorrect segmentation. In order to use this new map
for segmentation and obtain the ROI, the new map of motion vectors were converted to binary blobs. Furthermore, a few morphological processes were performed to enhance the segmentation stage. Finally, segmentation for each blob was performed and processed using the state-of-art PFF human detector.

**Figure 1** Proposed system

![Flowchart Diagram]

2.1 Optical flow

Optical flow can be used to obtain velocity measurements for object pixels but it cannot provide any information regarding the type of object. In other words, it cannot determine if an object is a human or car. There has been rapid progress in the field of object detection techniques. A few of these techniques can be run in real time while others cannot because of their complexity.

Optical flow is the pattern of apparent motion for objects, edges, and surfaces in a visual scene caused by the relative motion between an observer (an eye or a camera) and the scene (Huston et al., 2008). Optical flow algorithms differ from the temporal difference method and the background subtraction method. Optical flow algorithms have the ability to detect the outline of the complete moving target. Figure 2 defines the direction and magnitude of the optic flow at each location and also defines the length of each arrow (Huston et al., 2008).
Enhancing pedestrian detection using optical flow for surveillance

Figure 2  The optical flow experienced by a rotating observer (see online version for colours)

Source: Huston et al. (2008)

Figure 3  The relationship between the motion field and the optical flow field (see online version for colours)

Figure 3 defines the relationship between the motion field and the optical flow field (Lihizelnikmanor, 2008).

\[
\text{Motion Field} = \text{Real world 3D motion}
\]

\[
\text{Optical Flow Field} = \text{Projection of the motion field onto a 2D image.}
\]

In other words, optical flow is an approximation of the local image motion based upon local derivatives in a given sequence of images. Optical flow techniques, such as motion detection, object segmentation, time to collision, expansion calculations, motion encoding, and stereo disparity measurements utilise this motion of the object’s surfaces and edges. Optical flow constraints according to CeLiu (2011) and Moeslund et al. (2006) are shown in Figure 4.

Figure 4  2D motion of pixel
The calculations for optical flow are as follows:

\[ O = I(x + p, y + q) - H(x, y) \]
\[ \approx (xy) + I_x p + I_y q - H(x, y) \]
\[ \approx (I(xy) - H(x, y)) + I_x p + I_y q \]
\[ \approx I_x + I_y p + I_y q \]
\[ \approx I_x + \nabla I \left[ p, q \right] \] (1)

In the limit as \( u \) and \( v \) go to zero, this becomes exact

\[ O = U_i + \nabla I \left[ \frac{\partial x}{\partial t}, \frac{\partial y}{\partial t} \right] \] (2)

A2D and 3D Motion estimation of optical flow was proposed by Barron and Thacker (2005). The 2D motion equation assumes \( I(x, y, t) \) is the centre pixel in the \( n \times n \) neighbourhood and if moves by \( \delta_x, \delta_y \) in time \( \delta_t \) to \( x + \delta_x, y + \delta_y, t + \delta_t \).

Since, \( I(x, y, t) \) and \( (x + \delta_x, y + \delta_y, t + \delta_t) \) represents images of the same point, therefore:

\[ I(x, y, t) = (x + \delta_x, y + \delta_y, t + \delta_t) \] (3)

The small local translations provided \( \delta_x, \delta_y \) and \( \delta_t \), which are not too big. We can perform a first order Taylor series expansion related to \( I(x, y, t) \) in equation (1) to obtain the Tafazzoli and Safabakhsh (2006) equation as shown below:

\[ I(x + \delta_x, y + \delta_y, t + \delta_t) = I(x, y, t) + \frac{\partial I}{\partial x} \delta_x + \frac{\partial I}{\partial y} \delta_y + \frac{\partial I}{\partial t} \delta_t + H.O.T. \] (4)

where \( H.O.T. \) is the higher order terms, which we assumed were small terms that could be safely ignored. Using the above two equations we obtain the following results:

\[ \frac{\partial I}{\partial x} \delta_x + \frac{\partial I}{\partial y} \delta_y + \frac{\partial I}{\partial t} \delta_t = 0 \]

or

\[ \frac{\partial I}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial I}{\partial y} \frac{\partial x}{\partial t} + \frac{\partial I}{\partial t} \frac{\partial x}{\partial t} = 0 \]

and finally:

\[ \frac{\partial I}{\partial x} v_x + \frac{\partial I}{\partial y} v_y + \frac{\partial I}{\partial t} = 0 \]

Here

\[ v_x = \frac{\delta_x}{\delta_t} \quad \text{and} \quad v_y = \frac{\delta_y}{\delta_t} \]

are the \( x \) and \( y \) components of image velocity or optical flow and
Enhancing pedestrian detection using optical flow for surveillance

\[
\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial I}{\partial t}
\]

are the image intensity derivatives at \(x, y, t\). Typically, these partial derivatives are written as:

\[
I_x = \frac{\partial I}{\partial x}, \quad I_y = \frac{\partial I}{\partial y}, \quad I_t = \frac{\partial I}{\partial t}
\]

where \((I_x, I_y, I_t)\) represent intensity derivatives. This equation can be rewritten more compactly as:

\[
(I_x, I_y, V_x, V_y) = -I_t
\]

\[
\nabla I \hat{v} = -I_t
\]

where \(\nabla I = (I_x, I_y)\) is the spatial intensity gradient and \(\hat{v} = (V_x, V_y)\) is the image velocity or optical flow at pixel \((x, y)\) at time \(t\). \(\nabla I \hat{v} = -I_t\) is referred to as 2D motion.

### 2.2 Lucas-Kanade

The Lucas-Kanade method is commonly used as a differential method for optical flow estimation in computer vision (Lucas and Kanade, 1981). It uses least squares criterion to solve basic optical flow equations for all the pixels in a neighbourhood. It is a purely local method that assumes that the flow is constant in neighbourhoods local to the pixel under consideration. Its advantages include a rate that can make very fast calculations and accurate time derivatives. The disadvantage of the Lucas-Kanade method is the creation of errors regarding the boundaries of moving object (Felzenszwalb et al., 2010).

The Lucas-Kanade method assumes that the displacement of the image contents between two nearby frames are small and approximately constant within a neighbourhood of point \(p\) as shown below:

\[
I_x(p_1)V_x + I_y(p_1)V_y = -I_t(p_1)
\]

\[
I_x(p_2)V_x + I_y(p_2)V_y = -I_t(p_2)
\]

\[
\ldots
\]

\[
I_x(p_n)V_x + I_y(p_n)V_y = -I_t(p_n)
\]

where \(P_1, P_2, \ldots P_n\) are pixels inside the window. \(I_x(p_1), I_y(p_1), I_t(p_1)\) are the partial derivatives of the image \(I\) with respect to position \(x\), \(y\) and time \(t\), evaluated at the point \(Pn\) and at the current time. The equations can be written in matrix form:

\[
A v = b,
\]

where

\[
A = \begin{bmatrix}
I_x(p_1) & I_y(p_1) \\
I_x(p_2) & I_y(p_2) \\
\vdots & \vdots \\
I_x(p_n) & I_y(p_n)
\end{bmatrix}, \quad v = \begin{bmatrix}
V_x \\
V_y
\end{bmatrix}, \quad b = \begin{bmatrix}
-I_x(p_1) \\
-I_x(p_2) \\
\vdots \\
-I_x(p_n)
\end{bmatrix}
\]

(5)
As shown in equation (5), motion vectors can be found [Figure 5(b)] from the image [Figure 5(c)]. However, the vectors of the image are not same nor are they continuous. A few of the vector images can gather around an object and can be easily used to find the outline of the object. This can help us in segmentation stage. Other vectors are considered to be a kind of noises, which is one of the disadvantages of the Lucas-Kanade algorithm. These noises will be processed in the next two stages.

**Figure 5** The enhancement process, (a) original frame (b) Luckas-Kanade result (c) proposed enhancement result (d) the binary blobs result (see online version for colours)

2.3 Gaussian filter

As shown in Figure 5(b), the motion vectors map includes a great deal of noise due to the nature of the standard L-K algorithm. A Gaussian filter is considered to be optimal for image smoothing as its convolution with an image averages the pixels in the image, effectively decreasing the difference in value between neighbouring pixels. It is very effective for removing noise and is used as a linear low pass filter. The advantage of this filter is it is computationally efficient. The degree of smoothing is controlled by $\sigma$ (larger $\sigma$ for more intensive smoothing). The 2-D, Gaussian filter has the following form:

$$g(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}}$$  \hspace{1cm} (6)
The parameter $\sigma$ in equation (6) is a user defined value. It is equal to the standard deviation of the Gaussian filter, and it can be adjusted according to the desired distribution. It should be noted that as the parameter varies the size of the kernel must be adapted or else the kernel may exclude a sufficient number of elements to maintain the ‘bell shape’ of the distribution.

Figure 5(c) shows the resulting vectors map after applying this filter. As demonstrated by Figure 5(c), the filter removed noise and improved the overall map vectors that were used for blobs segmentation stated in next stage.

2.4 Morphorigical process

In this process, opening and closing morphorigical operations were performed to remove isolated noises as shown in Figure 6(a) to obtain $\text{Mo}(x, y, t)$. A threshold was then applied to obtain the binary blobs. Finally, the resulting refined image $\text{MR}(x, y, t)$, as shown in Figure 5(d) and Figure 6(b), that only contained the blobs (moving objects) that met the required conditions, was produced.

Figure 6 (a) Before the morphorigical process (b) After the morphorigical process

If an optical flow technique had been applied in the previous step, this phase would have performed an AND operation between the output images produced in the previous phases, $\text{Br}(r, c, t)$ and $\text{Mr}(r, c, t)$. The aim here is to take advantage of the optical flow information to improve PFF detector. Therefore, the results of L-K optical flow provided the detector with the blobs used as ROIs.

2.5 PFF detector

PFF is described as a complete learning-based system for detecting and localising objects and humans in images using combined multi-scale deformable part models. PFF has the ability to represent highly variable object classes and achieve state-of-the-art results in PASCAL object detection challenges. While deformable part models have become popular, their value had not been demonstrated using difficult benchmarks such as the PASCAL, of with PETS datasets. The PFF detector relies on new methods for discriminative training with partially labelled data. A margin-sensitive approach combined with data-mining hard negative examples and formalism called latent SVM. A
latent SVM is semi-convex and the training problem becomes convex once latent information is specified for the positive examples. This leads to an iterative training algorithm that alternates between fixing latent values for positive examples and optimising the latent SVM objective function (Felzenszwalb et al., 2010).

3 Results

An experiment was performed using a benchmark dataset (PETS, 2006) (Song et al., 2003) that contained 744 frames. This database has been recognised as a standard in video surveillance environments that use public spaces such as shopping malls, train stations, airports, and outdoor parks. Only outdoor scenes were used in this study because these frames contained non-controlled real situations in surveillance environments, with crowded and uncrowded sequences, lighting changes, and occlusions. However, the PETS database was designed for human detection and activity recognition for surveillance areas. Figure 5(a) shows an example of an original frame that is processed to obtain a vectors map using optical flow. Figure 5(b) shows the results of using a standard Lucas-Kanade optical flow technique where several noise vectors were shown. Figure 5(c) shows an enhanced vector maps. The proposed method significantly enhanced the Lucas-Kanade technique by filtering and smoothing unwanted vectors. Figure 5(d) shows the binary blobs, which were found throughout the new vector map. These binary blobs are used to obtain the ROI and used at as inputs for the detectors. Figure 7 shows the final detection results.

Figure 7 Final detections (see online version for colours)

For further evaluation, the performance of the proposed algorithm was compared with the PFF standard algorithm following a precision-recall curve. A precision-recall curve is a trade-off between precision and recall. Precision is the number of true positives divided by the total number of elements labelled as belonging to the positive class [equation (7)]. Recall is defined as the number of true positives divided by the total number of elements that actually belong to the positive class [equation (8)].


Enhancing pedestrian detection using optical flow for surveillance

Precision = \frac{TP}{TP + FB} \quad (7)

Recall = \frac{TP}{TP + FN} \quad (8)

Figure 8 shows the precision-recall curves obtained by the proposed and the PFF algorithms. The results show that the proposed algorithm performed better than the PFF algorithm as its precision values were higher than the precision values achieved by the standard PFF algorithm with changes in recall values.

An experimental test was performed for the proposed technique using PFF detector thresholds (0, –0.3, –0.5, –0.7, –1.0 and –1.5). The entire data set contained and used 744 frames. Table 1 compares the original PFF detector and the proposed method. Using the proposed method, the number of false positive (FP) decreased, which lead to higher precision and greater accuracy. Table 1 also shows that the processing time was reduced using the proposed method. Figure 8 shows the final result for the proposed method where each human in the scene was detected identified using red frames. These results will be used for further research in tracking and behaviour recognition.

Table 1  Comparison between the original PFF detector and the proposed method

<table>
<thead>
<tr>
<th>Threshold</th>
<th>nP</th>
<th>TP</th>
<th>FP</th>
<th>Precision</th>
<th>Recall</th>
<th>Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold = 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PFF detector – original</td>
<td>1,165</td>
<td>708</td>
<td>7</td>
<td>0.99</td>
<td>0.61</td>
<td>2.337*1,000</td>
</tr>
<tr>
<td>PFF detector with OF – proposed</td>
<td>1,165</td>
<td>634</td>
<td>8</td>
<td>0.987</td>
<td>0.54</td>
<td>1.86*1,000</td>
</tr>
<tr>
<td>Threshold = –0.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PFF detector – original</td>
<td>1,165</td>
<td>793</td>
<td>21</td>
<td>0.974</td>
<td>0.68</td>
<td></td>
</tr>
<tr>
<td>PFF detector with OF – proposed</td>
<td>1,165</td>
<td>733</td>
<td>16</td>
<td>0.978</td>
<td>0.63</td>
<td>1.54*1,000</td>
</tr>
<tr>
<td>Threshold = –0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PFF detector – original</td>
<td>1,165</td>
<td>851</td>
<td>48</td>
<td>0.947</td>
<td>0.73</td>
<td>3.203*1,000</td>
</tr>
<tr>
<td>PFF detector with OF – proposed</td>
<td>1,165</td>
<td>786</td>
<td>29</td>
<td>0.964</td>
<td>0.68</td>
<td>1.675*1,000</td>
</tr>
<tr>
<td>Threshold = –0.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PFF detector – original</td>
<td>1,165</td>
<td>900</td>
<td>102</td>
<td>0.898</td>
<td>0.77</td>
<td>2.272*1,000</td>
</tr>
<tr>
<td>PFF detector with OF – proposed</td>
<td>1,165</td>
<td>856</td>
<td>57</td>
<td>0.938</td>
<td>0.74</td>
<td>1.74*1,000</td>
</tr>
<tr>
<td>Threshold = –1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PFF detector – original</td>
<td>1,165</td>
<td>1,000</td>
<td>343</td>
<td>0.745</td>
<td>0.86</td>
<td></td>
</tr>
<tr>
<td>PFF detector with OF – proposed</td>
<td>1,165</td>
<td>941</td>
<td>195</td>
<td>0.828</td>
<td>0.81</td>
<td>1.76*1,000</td>
</tr>
<tr>
<td>Threshold = –1.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PFF detector – original</td>
<td>1,165</td>
<td>1079</td>
<td>2616</td>
<td>0.292</td>
<td>0.93</td>
<td>2.366*1,000</td>
</tr>
<tr>
<td>PFF detector with OF – proposed</td>
<td>1,165</td>
<td>1042</td>
<td>592</td>
<td>0.638</td>
<td>0.89</td>
<td>1.49*1,000</td>
</tr>
</tbody>
</table>

Notes: nP = total number of objects in ground truth, TP = true positive, FP = false positive.
The proposed algorithm was evaluated and its performance is shown in Table 1. Our implemented method had a maximum true detection range once we increase the threshold. Additionally, the FP for the proposed method was less than it was for the standard PFF detector. An average calculation was performed to determine the enhancement value for proposed method in term of accuracy. The results shown in Table 1 reveal that the average calculation was 118% better than the PFF standard.

Table 2 lists the execution times for each algorithm where the flow was calculated for six thresholds. Timing was measured using a computer (Intel 2.7 Ghz, Core i7 CPU with 8GB Ram) with Matlab.

<table>
<thead>
<tr>
<th>Threshold</th>
<th>PFF (sec)</th>
<th>Proposed method (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.337*1,000</td>
<td>1.86*1,000</td>
</tr>
<tr>
<td>-0.3</td>
<td>2.942*1,000</td>
<td>1.54*1,000</td>
</tr>
<tr>
<td>-0.5</td>
<td>3.203*1,000</td>
<td>1.675*1,000</td>
</tr>
<tr>
<td>-0.7</td>
<td>2.272*1,000</td>
<td>1.74*1,000</td>
</tr>
<tr>
<td>-1.0</td>
<td>2.296*1,000</td>
<td>1.76*1,000</td>
</tr>
<tr>
<td>-1.5</td>
<td>2.366*1,000</td>
<td>1.49*1,000</td>
</tr>
</tbody>
</table>

The results show that the proposed algorithm was much faster than the standard PFF detector. The proposed algorithm was almost twice as fast as the standard PFF detector, and thus can be used in real-time tracking applications. An average calculation was performed to obtain an enhancement value for the proposed method in terms of speed. As shown in Table 2, the results reveal that the proposed method was 37% faster than PFF standard.

4 Conclusions

In this paper, an optical flow algorithm was proposed for the purpose of detecting human figures. This algorithm contributes to this area of research by improving state-of-the-art
Enhancing pedestrian detection using optical flow for surveillance

detectors for surveillance applications. The algorithm was designed to work in real time (25 fps). Our results revealed that the proposed method is very practical as the incorporation of our proposed optical flow method into a state-of-the-art tracker greatly improve its robustness. The detector’s precision and accuracy increased by 118%. Compared to the standard PFF detector, the computational complexity of the proposed algorithm was greatly reduced. Accordingly, the processing time decreased by 37%. Therefore, it meets the requirements of real-time systems. Future work will focus on integrating the proposed method into tracking systems.

References


