DETECTION AND AUTOMATIC IDENTIFICATION OF HUMAN WALK

Josip Musić, Mojmir Cecić, Vlasta Zanchi
Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture
R. Boškovića b.b., Split
Croatia
jmusic@fesb.hr, mcecic@fesb.hr, vzanchi@fesb.hr

ABSTRACT
Use of video in robot vision and machine understanding is fundamental for a number of high-level applications. These include identification of humans by “the way they walk” (biometrics), human-robot interaction, pedestrian safety, automated video surveillance etc. For high-level procedures to be performed, low-level operations have to be executed. These include target detection, tracking and labeling as well as understanding of target interaction. There are several commercially available programs for detection and analysis of human walk like W4, Pfinder and Spfinder [1], [2]. Some of these programs use stereo imagery, which is not always suitable. Algorithm introduced in this paper doesn’t require stereo imagery. It uses nonparametric background modeling for target extraction and star-skeletonization for identification of target class. Obtained results are promising and can be used in further development.

KEY WORDS
Nonparametric modeling, star-skeletonization, automatic identification

1. Introduction

In robotics and video surveillance there is great demand for automated identification of moving objects, understanding of their actions and interaction with them. With that in mind one must extract all of the moving targets from video sequence obtained by automated system, discriminating between actual movement and movement in the background. This can be done by number of methods like background subtraction (parametric and nonparametric, simple subtraction etc.), temporal differencing, optical flow, etc. Once targets of interest are extracted from video sequence they have to be compared with known objects based on some distinct characteristics. In the case of human walk often-used characteristic is cyclic nature of human walk although there are other characteristics available.

In this paper combination of nonparametric background subtraction (for target extraction) and star-skeletonization (for identification of human walk) was used. This combination has been show to be very good. Results obtained by use of this combination are promising and can be used as basis for further development of the algorithm, which can be used as a first step for number of applications (biometrics and human-robot interaction being just couple of them).

2. Background modeling

The simplest way to detect moving object in video sequence is to compare each new frame of the sequence with know picture of the scene (or background). There are several ways to model the background. Depending on the chosen method for background modeling different results are obtained. The choice of method also affects how good the background model adapts to small changes in the scene (including illumination changes). In general background model must be invariant to these changes but adaptation should be made in such a way that even a slower moving targets could be detected. Described method is called background subtraction.

2.1 Background subtraction

There are two ways to approach background modeling in this method and in general. First there is a pixel value approach in which changes in value of a single pixel are detected. Secondly there is a block approach method, which detects value changes of entire block of pixels. We used pixel-based approach since it offers good granularity and can detect smaller targets (or targets with smaller number of pixels). There are several ways to extract pixel-based features that include: statistical modeling (parametric and nonparametric), hidden Markov models, extraction of edge features etc. We decided to choose statistical modeling based on several criteria: computational cost, obtained results and mathematical complexity. Statistical modeling can be divided into two main groups: parametric and nonparametric, main difference being necessity for additional calculation (algorithms) in parametric modeling. This fact was crucial in choosing nonparametric modeling [3], [4]. Nonparametric modeling estimates density function
directly from the given data. By doing this one, avoids to choose a model and distribution parameters. A particular nonparametric method is kernel density estimation. Here probability density function (pdf) can be calculated using

\[
\hat{p}(x) = \frac{1}{N} \sum_{i=1}^{N} K_{\sigma}(x - x_i)
\]

(1)

at value \(x\), where \(K_{\sigma}\) is a kernel function and \(\sigma\) is bandwidth of kernel function. If color video is available then it is necessary to use three-dimensional (because of RGB components) pdf estimation obtained as a product of three 1-D kernel estimates

\[
\hat{p}(x) = \frac{1}{N} \prod_{i=1}^{3} K_{\sigma_j}(x - x_{ij})
\]

(2)

where \(\sigma_j\) is bandwidth for each color.

Kernel density function \(K_{\sigma}\) can be one of several forms but in this paper Gaussian function is chosen so equation (2) becomes

\[
Pr(x_i) = \frac{1}{N} \prod_{j=1}^{3} \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-\frac{(x_{ij} - x_i)^2}{2\sigma_j^2}}
\]

(3)

When automated system provides one with sample of intensity values \(x_1, x_2, \ldots, x_N\) for certain pixel from video sequence, pdf for observed pixel can be calculated using (3). With pdf calculated we can estimate the probability that observed pixel belongs to that pdf. Calculated probability is then thresholded. Used threshold \(th\) is defined for entire picture. Pixel will be considered a part of moving object if

\[
Pr(x_i) < th
\]

(4)

To achieve adaptivity of a background model to slight changes in the background (like swinging of tree branches or even small movement of camera due to wind load) older intensity values in the sample are continually replaced with new ones. This replacement of sample values must be done in such a way and in such speed that slower moving targets can be detected. Rate of replacement is therefore a compromise between desired sensitivity of background model and expected target speed.

Important parameter to be calculated is kernel bandwidth \(\sigma_j\). Too small or too wide bandwidth can lead to undesired effects. Kernel estimate is calculated by median absolute deviation (\(m\)) for consecutive intensity values in the sample. Median is chosen for this calculation because of its low sensitivity to intensity value jumps. For this calculation it is assumed that pair of consecutive values \((x_i,x_{i+1})\) are from the same local-in-time distribution. This distribution is assumed to be Gaussian distribution \(N(\mu,\sigma^2)\) meaning that distribution of deviation \((x_i-x_{i+1})\) is also a Gaussian \(N(0,2\sigma^2)\). Because of the symmetric nature of Gaussian distribution, the median value of absolute deviation is equal to quarter percentile of deviation distribution

\[
P_r(N(0,2\sigma^2);m) = 0.25
\]

(5)

Kernel bandwidth \(\sigma_j\) is then calculated using

\[
\sigma = \frac{m}{0.68\sqrt{2}}
\]

(6)

2.2 Elimination of false detections

Used kernel density estimation method yields good detection results, but some false detections are also present. These false detections are due to small movement in the scene background, which were not present during background model generation, random noise or some other unpredictable source.

Fig. 1. (a) original image (b) first stage detection result

Elimination of false detections (or second stage of detection) is also done based on statistical modeling. If some pixel is detected as a part of a moving object, but in fact it is part of the background then that pixel will have high probability of belonging to background distribution of some pixel in the neighbourhood of observed pixel. This probability is called pixel displacement probability \(P_N(x_i)\) and is defined as probability that observed pixel value belongs to the distribution of some point in the neighbourhood \(N\) of observed pixel

\[
P_N(x_i) = \max_{y\in N(x_i)} Pr(x_i \mid B_y)
\]

(7)

where \(B_y\) is sample of intensity values for pixel \(y\) and \(Pr(x_i \mid B_y)\) is calculated using (3). Once pixel displacement probability \(P_N(x_i)\) is obtained it is thresholded. Used threshold is often the same as the threshold \(th\) used in first stage of detection. Unfortunately this thresholding also eliminates some true detections, so another constraint is needed. Thus component displacement probability \(P_C\) is introduced. This probability is estimated using:
\[ P_C = \prod_{x \in C} P_N(x) \quad (8) \]

Assumption behind introduction of \( P_c \) is that all pixels comprising detected object will move between consecutive frames and not only some of them. Results obtained using (7) are then thresholded with \( th_2 \). Pixel will be considered a part of the background when

\[ (P_N(x) > th_1) \land (P_C(x) > th_2) \quad (9) \]

### 2.3 Color video

Implementation of color video in automated systems has several advantages over monochromatic video. Most important is shadow removal [5]. Shadows represent big problem in moving object detection and identification since they are often mistaken for real targets. They also distort shape of the real targets, which can lead to wrong identification results. Shadow removal based on color can be partial but sufficient enough to be considered an advantage. To that end chromaticity \( (r,g,b) \) coordinates are introduced [6]. These coordinates lower sensitivity to shadows and are calculated using

\[
\begin{align*}
r &= \frac{R}{R+G+B} \\
g &= \frac{G}{R+G+B} \\
b &= \frac{B}{R+G+B}
\end{align*}
\quad (10)
\]

where \( r+g+b=1 \).

By introducing rgb coordinates we have moved from three-dimensional space (RGB cube) to two-dimensional space \( (r+g+b=1) \) plane. Thus only two out of three chromaticity coordinates are needed. Introduction of these new coordinates causes loss of lightness information. This effect is eliminated by implementation of lightness parameter \( s \)

\[ s = R+G+B \quad (11) \]

Reduction of shadow effects is done by selecting a subset \( S \) of extracted sample that satisfy following

\[ S = \left\{ x_i \mid x_i \in B \land \alpha \leq \frac{s_i}{s} \leq \beta \right\} \quad (12) \]

where \( \alpha \) and \( \beta \) are obtained by experimentation.

Examples of described detections of first and second stage are show in next figure

![Fig.2. (a) first stage detection results (b) second stage detection results (c) results after use of morphological operations and median filtering](image)

### 3. Star-skeletonization

After detection stage of an algorithm, identification is done. There are several methods available, but in this paper we chose star-skeletonization [7]. Star-skeletonization has several advantages over other methods:

- it is computationally cheap (no iterations are required)
- doesn’t require cardboard human model
- it can easily be upgraded for other objects (animals, cars etc.)

It is also worth noticing that changing cut-off frequency of low pass filter one changes sensitivity of star-skeleton, and by doing that influences number of external points of the skeleton.

Since no detection method is perfect (spur pixels and “holes” in detected objects are present) certain morphological operations need to be performed before actual identification can be done. These operations include erosion and dilatation and also no-morphological operation of median filtering [8], [9].

Star-skeleton of a detected moving object is extracted in several steps:

1.) Calculation of centroid coordinates \((x_c,y_c)\) of a silhouette of moving object using

\[
\begin{align*}
x_c &= \frac{1}{N_s} \sum_{i=1}^{N_s} x_i \\
y_c &= \frac{1}{N_s} \sum_{i=1}^{N_s} y_i
\end{align*}
\quad (13)
\]

where \((x,y)\) are coordinates of pixels comprising the silhouette and \( N_s \) is the number of all pixels in silhouette.

2.) Calculation of the distance between every pixel \((x_i,y_i)\) and centroid using
\[ d_i = \sqrt{(x_c - x_i)^2 + (y_c - y_i)^2} \]  \quad (14)

3.) Smoothing of the function \( d_i \) by means of low-pass filter in Fourier domain. Filter output is function \( \hat{d}_i \).

4.) Finding local maxims of the function \( \hat{d}_i \) and related silhouette pixel coordinates.

5.) Connecting found external points with centroid, thus obtaining star-skelet.

Extracted star-skelet is then used for calculation of angle between vertical line and leading leg (there is no discrimination between left and right leg) as well as angle between torso and vertical line. In the paper we assumed that object is always moving from right to left, but generalization of the method can easily be done by tracking x-coordinate of centroid in several consecutive frames.

\[ \Theta = \arctg \frac{l_x - x_c}{l_y - y_c} \]  \quad (15)

where \((l_x, l_y)\) are coordinates of extreme points of the skelet.

3.1 Object identification

For object identification, angle between leading leg and vertical line is used.

Angle between torso and vertical line can be used to distinguish walk from running as shown in [7]. Basis for human identification used here is cyclic nature of human walk. To be more precise identification is done by calculating frequency of human walk in video sequence and comparing it with known data from literature. As a restraint calculated angles are inspected for “abnormal” values.

Actual identification has two pre-processing steps. Firstly obtained angle function is passed through filter with transfer function defined in z-domain as

\[ H(z) = 1 - az^{-1} \]  \quad (16)

where value for \( a \) is obtained experimentally. Fujiyoshi [7] demonstrated that \( a=1 \) is acceptable value.

This filter is used to eliminate DC component and low frequency noise components in power spectrum. As a second step output from filter is then used to calculate autocorrelation. Autocorrelation is performed in order to emphasize major cyclic component.

With pre-processes over we can now move into Fourier domain and analyze power spectrum of obtained signal. Analyses include calculation of frequency of a component with biggest power spectrum value. This frequency is then thresholded (with threshold being double frequency of a normal human walk) and a decision is made whether moving object in video sequence is a walking human or not.

Final identification is not based solely on calculated frequency, but also on results of inspection for “leading leg angles”. This is done because one can calculate...
correct frequency of human walk, but with unacceptable angle values.

4. Materials and methods

Experimentation was carried out using SONY DCR-TRV110E video camera recorder, tripod and a computer (Pentium Celeron 1.1GHz, 256MB RAM) with video capture card. Video sequences were recorded on SONY HMP (Metal Particle) Hi8 video tape. Once recorded, sequences were transferred to hard drive (in AVI file format) using VidCap32 software. During this process suitable compression method was chosen, or to be more precise it was decided that no compression (full uncompressed) would be used. Thus better quality video was obtained and since hard drive space isn’t an issue, decision was logical. Duration of video was adjusted to include only actions of interest and video format was chosen so it was in accordance with MATLAB requirements. All this was done with VideoMach 2.7.2 software. With this pre-processing of video over, algorithm for detection and identification could be used.

Special care was given to making videos. Two types of video sequences were made: indoor and outdoor. During video sequence acquisition few hypotheses were assumed. Moving object of interest is the biggest moving object in the scene (there are no occlusions between different moving objects). No occlusion of the moving object and the scene was allowed. Subjects moved from the left to the right of the scene. Also great care was given to lightning conditions (for example position of the Sun in outdoor scenes). These restrictions during video sequence acquisition are necessary because of the way the algorithm works at present time.

5. Experimental results

This section presents the results of the proposed algorithm. Used video sequences were of 320x240 resolution. Proposed algorithm was tested on 10 video sequences recorded in two different scenes (one outdoor and one indoor scene). Used sequences contained human and non-human moving objects. Detection results are evaluated subjectively by visual means, while results of automatic detection can be easily confirmed or dismissed. Summary of obtained results is shown in next figure

<table>
<thead>
<tr>
<th>Sequence number</th>
<th>Sequence description</th>
<th>Type of the scene</th>
<th>Detection results</th>
<th>Identification results</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>male 1 walking</td>
<td>outdoor</td>
<td>good</td>
<td>correct</td>
</tr>
<tr>
<td>2.</td>
<td>male 2 walking</td>
<td>outdoor</td>
<td>good</td>
<td>correct</td>
</tr>
<tr>
<td>3.</td>
<td>car passing by</td>
<td>outdoor</td>
<td>good</td>
<td>correct</td>
</tr>
<tr>
<td>4.</td>
<td>male 1 walking</td>
<td>indoor</td>
<td>very good</td>
<td>incorrect</td>
</tr>
<tr>
<td>5.</td>
<td>male 2 walking</td>
<td>indoor</td>
<td>very good</td>
<td>correct</td>
</tr>
<tr>
<td>6.</td>
<td>female 1 with skirt</td>
<td>indoor</td>
<td>very good</td>
<td>correct</td>
</tr>
<tr>
<td>7.</td>
<td>female 2 walking</td>
<td>indoor</td>
<td>very good</td>
<td>incorrect</td>
</tr>
<tr>
<td>8.</td>
<td>male 1 carrying object</td>
<td>indoor</td>
<td>very good</td>
<td>incorrect</td>
</tr>
<tr>
<td>9.</td>
<td>male 1 walking with slight change in speed</td>
<td>indoor</td>
<td>very good</td>
<td>correct</td>
</tr>
<tr>
<td>10.</td>
<td>male 1 with trousers with similar color to the background</td>
<td>indoor</td>
<td>poor</td>
<td>incorrect</td>
</tr>
</tbody>
</table>

Table 1. Result summary

Some results in the Table 1. warrant additional explanation. In sequence number 10 subject in indoor scene wore trousers of similar color to the color of the background. This made detection of the subjects legs impossible with selected threshold. Thus automatic detection results were compromised. In sequence number 6 female subject wore skirt and even then algorithm performed well. In sequences number 4 and 7 detection algorithm produced good results but automatic identification part of the algorithm misclassified object as non-human. Reason for this was the fact that subjects natural walk frequency was higher than used average. This can be countered by additional restraints and taking into account possibility of person running. In sequence number 8 subject was carrying load that caused the misclassification of object class. In all other sequences both detection and identification were done correctly and without any problems.

6. Conclusion

In this paper automatic detection and identification method was described and tested. Although here presented theoretical background is well known we introduced some improvements in our algorithm and added some additional parts to improve it. Used algorithm was written in MATLAB 6.12 programming environment.
During programming process special attention was not given to the computational cost because the intention was to show performance (not computational) of proposed method. Speed up techniques can be found in [3],[4],[10],[11]. In previous section advantages and disadvantages of the algorithm were mentioned but we shall restate them here for the sake of completeness. Algorithm can adapt to slight changes in the background scenery as well as some light changes (not sudden). When all hypotheses were obeyed obtained results were good. Identification is done in robust way because the detection results don’t have to be perfect (meaning some errors in object detection are allowed) for it to work. Some problems are present when object is small and when subjects color scheme is similar to the one of the background. Computational cost in present form can also be considered a major drawback in real-time usage. Future work should include the following. Algorithm should be able to work in scenes with multiple moving objects (with occlusion) and more complex scenery. Also suitable automatic threshold selection techniques that take into account needed sensitivity of observed part of the image need to be developed. This is to avoid cases when, due to similarity of the background and moving object colors, detection isn’t possible. In the identification part of the algorithm additional conditions and restraints need to be implemented. To summarize, used algorithm has great potential for many applications (from robot human interaction to human identification using walk as biometrics) but additional development is needed.

References: