

Human Motion Detection Using Fuzzy Rule-base Classification Of Moving Blob Regions

John See¹, Lee Sze Wei², Madasu Hanmandlu³

¹Center of Biometrics and Bioinformatics, Faculty of Information Technology
Multimedia University, 63100 Cyberjaya, Selangor, Malaysia
E-mail: johnsee@mmu.edu.my

²Center of Multimedia Communication, Faculty of Engineering
Multimedia University, 63100 Cyberjaya, Selangor, Malaysia
E-mail: swlee@mmu.edu.my

³Department of Electrical Engineering
I.I.T. Delhi, Hauz Khas, New Delhi 110016, India
E-mail: mhmandlu@ee.iitd.ernet.in

Abstract

The task of detecting and classifying human motion is an important preliminary tool for many high-level applications. However, many approaches suffer from the lack of robust classification and proper motion cues. This paper presents a novel human motion detection algorithm that uses a fuzzy rule-base classification scheme based on moving blob regions. This approach first obtains a motion image through the acquisition and segmentation of video sequences. Then, preprocessing is applied to the motion image before major blobs are identified. Using motion estimation and ellipse fitting, three blob characteristics are extracted from the major blobs as classification criteria. These characteristics are used as inputs to a fuzzy rule-base for classification of the detected motion. Through experimental evaluation – a database test and real-time field test, the implemented system achieved good detection rates in both tests at efficient real-time speeds. In comparison with earlier approaches, this algorithm also managed better detection rates. Above all, the performance of the proposed algorithm has demonstrated its feasibility for an effective real-time implementation.

Keywords:

Human motion detection, human motion analysis, motion classification, fuzzy rule-base, security monitoring.

1. Introduction

Generally, human motion detection is the process of recognising a human being based on certain feature cues or patterns exhibited by the detected motion. In the rich area of human motion analysis, the task of detecting human motion is often a useful tool for higher-level vision applications such as human body tracking, gait recognition and activity recognition.

In human motion analysis, the perceptions of human motion are many. Johansson, in an early pioneering experiment on moving light displays (MLD), perceived the motion of the

human body on a basis of a simple stick model [1], which describes the body part segments of the head, torso, arms and legs. However, the task of locating human body parts as feature cues can be quite computationally expensive.

As highlighted in [2], the more efficient notion will be to detect the presence of a human being without having to pre-determine its body or limb segments. Polana and Nelson [3] were among the first to champion the idea of using low-level visual features to track human motion. In their own words, they proposed a way to “get your man without finding his body parts”.

Blobs are defined as a group of pixels that belong to a similar object in motion. They have proven to be a better feature cue than points, corners or ridges as they usually have a larger coverage area and total occlusion of the subject is more unlikely to happen. Rossi and Bozzoli [4] successfully used moving blobs to track and count people crossing the field of view of a vertically mounted camera. In a different approach with blobs, Bregler [5] represented each pixel in each motion image by its optical flow characteristics. These flows are then grouped into blobs that have coherent motion and are characterised by a mixture of multivariate Gaussians. Here, optic flow is useful to characterise each moving pixel according to certain features of the flow vector.

The task of detecting human motion is incomplete without the classification phase to distinguish human movements from other motions belonging to animals and objects. With the emerging use of fuzzy logic in various applications, fuzzy-based classification schemes [6] have also proven to yield better accuracy rates than conventional shape-based [7] and motion-based [8] techniques.

In this paper, a novel human motion detection algorithm that uses a fuzzy rule-base classification scheme based on moving blob regions is proposed. The algorithm flow is presented in Section 2 while further details of the proposed algorithm are described in Section 3. Experimental results are shown in Section 4 while comparative analysis and further discussion are presented in Section 5. Finally, Section 6 concludes this paper.

2. Algorithm Flow

The proposed algorithm consists of five stages – image acquisition and segmentation, preprocessing, major blob identification, blob characteristic extraction and motion classification. Figure 1 shows the process flow of the proposed human motion detection algorithm. Each of these stages will be described in detail in Section 3.

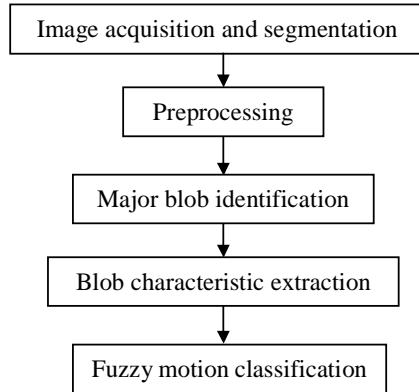


Figure 1 – Proposed algorithm flow

3. Human Motion Detection

3.1 Image Acquisition and Segmentation

Image acquisition is a common preliminary step in any motion-based vision application to obtain image frames from a stationary or moving camera, or multiple cameras. Usually, a frame grabber is used to subsample a sequence of video images at a certain frame rate before the actual processing begins.

Generally, video sequences can be of high frame rate (10-30 *fps*) or low frame rate (< 10 *fps*)¹. It is crucial to take into consideration the type of video system used so that the appropriate segmentation method can be implemented.

After the acquisition of image frames, image segmentation can be performed using two different methods – adjacent frame subtraction and background subtraction, depending on the frame rate of the video sequences.

Frame segmentation

For high frame rate sequences, the adjacent frame subtraction method is used since the change of motion between consecutive frames is very small. Thus, the difference image, $D(x,y,t)$ between an input frame $F(x,y,t)$ and the next acquired frame $F(x,y,t+c)$ after a fixed time interval c is given by

$$D(x, y, t) = abs\{F(x, y, t) - F(x, y, t+c)\} \quad (1)$$

The more widely known background subtraction technique is used for low frame rate sequences where the change of

¹ *fps* denotes frames-per-second, the standard measure of frame rate

motion is larger. This method eliminates the stationary background, leaving only the desired motion regions. The difference image, $D(x,y,t)$ between an input frame $F(x,y,t)$ and the background image $B(x,y,t)$ is given by

$$D(x, y, t) = abs\{F(x, y, t) - B(x, y, t)\} \quad (2)$$

In certain video systems, the acquisition process may be susceptible to erratic changes in illumination, reflection, and noise. To reduce the sensitivity to intensity changes caused by these factors, Gaussian spatial filtering can be applied before performing differencing.

Finally, a *motion image* is produced by thresholding the difference image $D(x,y,t)$ by a certain threshold level λ_D .

3.2 Preprocessing

The preprocessing stage consists of two tasks – morphological operations and blob labeling, and they are intended to prepare the motion image for the blob identification stage.

Morphological operations

The following morphological operations are performed on the motion image:

- Closing – Morphological closing smoothens sections of contours, fuse together narrow breaks and long gulfs.
- Fill holes – A flood-fill operation is then performed to close up the remaining small holes.
- Removal of motion at boundary – Pixels of the motion region that are located along the boundary are eliminated to avoid ambiguity of the region belonging to a possible moving object.

Blob labeling

The final motion image contains many *blobs*, or regions of detected motion. These motion blobs are counted and labeled individually, and their respective sizes are also determined.

3.3 Major Blob Identification

In this stage, the identification of a *major blob* is an essential step towards determining potential human motion. A *major blob* is defined as a motion blob that shows potential area size to be considered a motion that is exhibited by a moving person. Before that, some anthropometric assumptions are used in the estimation of the area size.

Anthropometrically, the average cross-sectional area of a human is estimated as $A_{Hm} = 0.5m^2$. Assuming that the scene length is perpendicular to the camera projection axis, this estimated area can be projected as

$$A_{Hp} = \frac{A_{Hm} D_{Sp}^2}{D_{Sm}^2} \quad (3)$$

in pixel square, where D_{Sm} and D_{Sp} are the real scene length (m) and its equivalent measure in pixels respectively.

From all the earlier labeled motion blobs, blobs that possess region areas that are smaller than the object blob cutoff level (default is 10% of the scene length) are filtered away first. Selection of major blobs is then performed by choosing only blobs that are more than half the estimated cross-sectional area of an average human, A_{Hp} .

If Area (OB_{*i*}) > $A_{Hp} / 2$, **then** blob_{*i*} is chosen as (MB)_{*j*}

for the *i*th object blob and *j*th major blob

(OB: object blob; MB: major blob)

3.4 Blob Characteristic Extraction

After the selection of major blobs, certain characteristics or feature cues are extracted to provide sufficient discrimination between human motion and motion caused by other objects.

Motion estimation

A method that uses motion blobs as features for the computation of motion flow (using local motion vectors) which was adapted from a similar approach in [9] is implemented. Here, local motion vectors are estimated from each successive pair of frames $F(x, y, t)$ and $F(x, y, t+1)$, based on a quasi-cross-correlation within a small W -by- W aperture (window) in the blobs. These vectors are used as feature correspondence for motion estimation, a concept first introduced by Shio and Sklansky [10].

The local motion vector is given by

$$\vec{\delta} = (\delta_x, \delta_y) = \left\{ (m, n) \mid \min_{m, n \in \{0, \dots, \pm l\}} \{C(m, n)\} \right\} \quad (4)$$

where

$$C(m, n) = \sum_{x=-1}^1 \sum_{y=-1}^1 |F(x, y, t+1) - F(x+m, y+n, t)| \quad (5)$$

and the estimated motion direction is restricted to an integer value between $-l$ and l for each location (m, n) . The average local motion vector of the *i*th object blob in frame $F(x, y, t)$ at instance t can be computed as

$$\vec{\delta}_i = -\frac{1}{P_i} \left(\sum_{p_x=1}^{P_i} (\vec{\delta}_x)_{p_x}, \sum_{p_y=1}^{P_i} (\vec{\delta}_y)_{p_y} \right) \quad (6)$$

for all I object blobs, where the *i*th blob contains P_i pixels.

In certain cases, rigidly thresholded motion regions may contain disconnected blobs (*minor* blobs) that belong to one similar object. In such cases, a simple region grouping method using centroid distance is employed to merge these minor blob regions with a certain major blob.

If distance{centroid (MIB)_{*i*} – centroid (MB)_{*j*}} < λ_{cd} , **then**

(MIB)_{*i*} is grouped with (MB)_{*j*}

for the *i*th object blob and *j*th major blob

(MIB: minor blob; MB: major blob)

From motion estimation, two blob characteristics can be extracted:

- Motion vector, which describes the overall motion orientation of the object of interest and it is given by

$$\vec{\delta}_j = \frac{1}{I_j} \left(\sum_{i(x)=1}^{I_j} \vec{\delta}_{i(x)}, \sum_{i(y)=1}^{I_j} \vec{\delta}_{i(y)} \right) \quad (7)$$

for all J major blobs, where the *j*th major blob contains I_j object blobs, and $\vec{\delta}_{i(x)}, \vec{\delta}_{i(y)}$ are the x and y components of each *i*th object blob respectively.

- Texture weight, which describes the average normalized gray scale value of the newly combined major blob region, which is given by

$$\zeta_j = \frac{\sum_{q_j=1}^{Q_j} (F(x, y, t))_{q_j}}{Q_j} \quad (8)$$

for all I object blobs, where the *j*th major blob contains Q_j pixels.

Ellipse fitting

The third blob characteristic, ellipse coverage area is derived from ellipse fitting. Here, a brute-force technique is used to fit a major blob into an elliptical region, taking into consideration various scale and tilt variations to cope with various gait postures of a human.

The percentage of major blob covered by the area of the ellipse, %CA is determined as

$$\%CA = \frac{\text{Area of major blob covered by ellipse}}{\text{Area of ellipse}} \times 100\% \quad (9)$$

The ellipse with the largest major blob coverage area is taken as the best-fit ellipse.

3.5 Fuzzy Motion Classification

In the final stage, a fuzzy rule-base motion classification approach to classify the extracted major blob is proposed. Three input variables which represent the classification criteria – motion vector distance, texture weight change, and percentage of ellipse coverage area, are formed from the three blob characteristics determined earlier. Meanwhile, one output variable is used to denote the confidence value of the classification decision.

A 5-set fuzzy IF-THEN rule base is constructed for a Mamdani inference system with trapezoidal membership functions. Figure 2 shows the fuzzy IF-THEN rule base with three input variables (X_1, X_2, X_3) and one output variable Y .

| | |
|--|------|
| Rule R_1 : If X_1 is NEAR and X_2 is SMALL and X_3 is UNFIT then Y is REJECT | ELSE |
| Rule R_2 : If X_1 is NEAR and X_2 is BIG and X_3 is UNFIT then Y is REJECT | ELSE |
| Rule R_3 : If X_1 is FAR and X_2 is SMALL and X_3 is UNFIT then Y is REJECT | ELSE |
| Rule R_4 : If X_1 is FAR and X_2 is BIG and X_3 is UNFIT then Y is REJECT | ELSE |
| Rule R_5 : If X_1 is NEAR and X_2 is SMALL and X_3 is FIT then Y is ACCEPT | |

Figure 2 – The 5-set fuzzy IF-THEN rule base

These rules are aggregated with minimum implication operator and defuzzified using centroid defuzzification to

obtain an output confidence value. A threshold level of 0.9 is set to determine the final decision to accept or reject the blob as a human motion.

4. Experimental Results

The proposed algorithm was assessed in two experimental settings – the database test and real-time field test.

4.1 Database Test

The database test is a compound database that comprises of two sub-databases:

- Database I, which consists of video sequences of human movements in various gait poses, non-human object motions and a combination of human and object movements, constructed from our motion capture system
- Database II, which consists of video sequences of three different camera angles (normal, oblique and normal-elevated) of a subject (in motion) in various conditions (clothing, objects carried, walking speed), taken from the Southampton Human ID Gait Database [11].

For Database I, a detection rate of 84% was achieved with optimum pre-set threshold parameters: object blob cutoff = 10%, body area threshold = $0.2m^2$ and motion image threshold, $\lambda_D = 0.10$. On the other hand, a perfect 100% detection rate was obtained for Database II. With the combination of both databases, the overall detection rate of the database test is 92%, as shown in Table 1.

Table 1 – Overall detection rate of the database test

| Database | Correct detection | Incorrect detection | Detection rate (%) |
|------------------------|-------------------|---------------------|--------------------|
| Database I | 21 | 4 | 84% |
| Database II | 25 | 0 | 100% |
| Combined | 46 | 4 | |
| OVERALL DETECTION RATE | | | 92% |

4.2 Real-time Field Test

In the real-time field test, a simple laboratory setting was used to detect human movements in a real-time scenario. Here, a different set of threshold parameters is used to suit the conditions: object blob cutoff = 20%, body area threshold = $0.2m^2$ and motion image threshold, $\lambda_D = 0.20$.

In performance evaluation, the detection rate, DR is taken as

$$DR = \frac{\text{Persons detected} - \text{FRP} - \text{FDO}}{\text{Persons counted}} \times 100\% \quad (10)$$

where FRP denotes Falsely Rejected Persons and FDO denotes Falsely Detected Objects.

In a preliminary motion-free test, the system took 616 seconds to process 1,000 image frames on an Intel Pentium IV 2.63Ghz, 504MB RAM system, which is approximately 1.62 *fps*. MATLAB[®] was used as the implementation tool.

The real-time test was left to run for a duration of about 2 hours and it achieved a good detection rate of 93.6% as shown in Table 2, which is better than the earlier database test result (92%). Frame acquisition was ran continuously with a short capture interval of about 0.6 seconds by utilising simple distributed processing (with dual processors). This allows the motion processing to be performed by a secondary dedicated processor. With this implementation, a successful detection process only took about 2-4 seconds on average.

Table 2 – Detection rate of the real-time field test

| | | |
|---------------------------------|---------------|--|
| PERSONS COUNTED | 47 | |
| • Falsely rejected person (FRP) | 2 | |
| • Falsely detected object (FDO) | 1 | |
| TOTAL INCORRECT DETECTION | 3 | |
| CORRECT DETECTION | 44 | |
| DETECTION RATE (%) | 93.75% | |

5. Comparative Analysis and Discussion

In comparison with earlier techniques that are of almost similar approach, the proposed algorithm performed better in terms of detection accuracy rate. Lipton et. al. [7] used a simple multi-feature classification metric with various blob properties as features, at it yielded only a detection rate of 82.8%. Meanwhile, Li and Leung [6] obtained a 92.6% detection rate using vertical projection feature cues with a fuzzy rule-based classification approach. On the other hand, the proposed algorithm is able to combine the robustness of fuzzy rule-base technique in decision making, and the effective use of moving blob regions as motion cues.

Table 3 – Comparison of detection accuracy rates with earlier techniques of similar approach

| Algorithm | Detection accuracy (%) |
|---|------------------------|
| Multi-feature classification metric (dispersedness, area of image blob) [7] | 82.80% |
| Fuzzy rule-based reasoning for classification from the vertical projection of motion cues [6] | 92.60% |
| Fuzzy rule-based classification of moving blob regions | 93.75% |

In further work, there are many aspects that can be improved. In view of the drawbacks of the proposed algorithm, improvements can be made to allow more robust and accurate interpretation of the blobs. The threshold parameters can be improved to be more adaptive to system noise and various image conditions (lighting, background). A more robust and complex ellipse fitting method can be use instead to add flexibility to its shape structure, but with consideration to computation speed as well.

In view of our experiments, this system can be further tested with a larger database size with a variety of poses, actions,



Figure 3 – A sample sequence from the Southampton Human ID database showing various stages of the proposed algorithm; original frame with motion of a person walking (top left); adjacent frame subtraction (top middle); motion image (top right); motion image after preprocessing (bottom left); motion vector $\vec{\delta}_j$ (bottom middle); and ellipse fitting (bottom right).

lighting conditions and camera angles. Both indoor and outdoor scenes can also be used to test the adaptability of the algorithm to different environment settings.

6. Conclusion

This paper presents a novel approach to the detection of human motion using a fuzzy rule-base classification scheme with moving blob regions as feature cues. In the proposed algorithm, three blob characteristics are extracted using motion estimation and ellipse fitting as criteria for classifying a detected motion. Moreover, this underlines the significance of using blobs as the feature cue of interest. The choice of using a fuzzy rule-base allows a more robust classification than most conventional methods.

Experimental results in both tests (database and real-time) have shown good achievement in both detection rate and speed. In comparison with earlier approaches that are similar in nature, the proposed algorithm yielded better detection rates than its predecessors. Above all, the performance of the proposed algorithm has also demonstrated its feasibility for an effective real-time implementation. In future work, there are many avenues for improvement on both algorithmic and experimental aspects of the algorithm.

Acknowledgments

The authors would like to thank the MMU Internal Funding and IRPA Grant 04-99-01-0068-EA064 for supporting the ASYMUTH project. Special thanks to Prof. Mark Nixon for the Southampton Human ID Gait Database [11].

References

[1] Johansson, G. 1975. Visual Motion Perception. *Sci. American* 232(6):76-88.
 [2] Aggarwal, J.K. and Cai, Q. 1997. Human Motion

Analysis: A Review. In Proceedings of IEEE Non-rigid and Articulated Motion Workshop, 90-102.
 [3] Polana, R. and Nelson, R. 1994. Low Level Recognition of Human Motion (or how to get your man without finding his body parts). In Proceedings of IEEE Workshop on Motion of Non-Rigid and Articulated Objects, 77-82.
 [4] Rossi, M., and Bozzoli, A. 1994. Tracking and Counting People. In 1st Int. Conf. on Image Processing, 212-216.
 [5] Bregler, C. 1997. Learning and Recognizing Human Dynamics in Video Sequences. In Proc. of IEEE CS Conf. on Comp. Vision and Patt. Recognition, 568-574.
 [6] Li, L., and Leung, M. K. H. 2001. Fusion of Two Different Motion Cues for Intelligent Video Surveillance. In Proc. of IEEE Int. Conf. on Electrical and Electronic Tech. (TENCON), Vol. 1, 341-344.
 [7] Lipton, A. J., Fujiyoshi, H., and Patil, R. S. 1998. Moving Target Classification and Tracking From Real-time Video. In Proc. of IEEE Workshop on App. of Comp. Vis., 8-14.
 [8] Cutler, R., and Davis, L. S. 2000. Robust Real-time Periodic Motion Detection, Analysis, and Applications. *IEEE Trans. Patt. Anal. Mach. Intell.* 22(8):781-796.
 [9] Horn, B.K.P. and Schunck, B.G. 1981. Determining Optical Flow. *Artificial Intelligence* 17:185-203.
 [10] Shio, A. and Sklansky, J. 1991. Segmentation of People In Motion. In Proc. of IEEE Workshop on Visual Motion, 325-332.
 [11] Shutler, J.D., Grant, M.G., Nixon, M.S. and Carter, J.N. 2002. On A Large Sequence-Based Human Gait Database. In Proc. of the 4th Int. Conf. on Recent Advances in Soft Computing, 66-71.

