Real-time human motion analysis by image skeletonization

Hironobu Fujiyoshi Alan J. Lipton

The Robotics Institute. Carnegie Mellon University.
5000 Forbes Avenue, Pittsburgh, PA, 15213
email: {hironobu|ajl}@cs.cmu.edu
URL: http://www.cs.cmu.edu/~ vsam

Abstract

In this paper, a process is described for analysing the motion of a human target in a video stream. Moving targets are detected and their boundaries extracted. From these, a "star" skeleton is produced. Two motion cues are determined from this skeletonization: body posture, and cyclic motion of skeleton segments. These cues are used to determine human activities such as walking or running, and even potentially, the target's gait. Unlike other methods, this does not require an a priori *human model, or a large number of "pixels on target". Furthermore, it is computationally inexpensive, and thus ideal for real-world video applications such as outdoor video surveillance.*

1 Introduction

Using video in machine understanding has recently become a significant research topic. One of the more active areas is activity understanding from video imagery [7]. Understanding activities involves being able to detect and classify targets of interest and analyze what they are doing. Human motion analysis is one such research area. There have been several good human detection schemes, such as [8] which use static imagery. But detecting and analyzing human motion in real time from video imagery has only recently become viable with algorithms like *Pfinder* [10] and W^4 [5]. These algorithms represent a good first step to the problem of recognizing and analyzing humans, but they still have some drawbacks. In general, they work by detecting features (such as hands, feet and head), tracking them, and fitting them to some *a priori* human model such as the *cardboard model* of Ju *et al* [6].

There are two main drawbacks of these systems in their present forms: they are completely human specific, and they require a great deal of image-based information in order to work effectively. For general video applications, it may be necessary to derive motion analysis tools which are not constrained to human models, but are applicable to other types of targets, or even to classifying targets into different types. In some real video applications, such as outdoor surveillance, it is unlikely that there will be enough "pixels on target" to adequately apply these methods. What is required is a fast, robust system which can make broad assumptions about target motion from small amounts of image data.

This paper proposes the use of the "star" skeletonization procedure for analyzing the motion of targets - particularly, human targets. The notion is that a simple form of skeletonization which only extracts the broad internal motion features of a target can be employed to analyze its motion.

Once a skeleton is extracted, motion cues can be determined from it. The two cues dealt with in this paper are: cyclic motion of "leg" segments, and the posture of the "torso" segment. These cues, when taken together can be used to classify the motion of an erect human as "walking" or "running".

This paper is organized as follows: section 2 describes how moving targets are extracted in real-time from a video stream, section 3 describes the processing of these target images and section 4 describes human motion analysis. System analysis and conclusions are presented in sections 5 and 6.

2 Real-time target extraction

The initial stage of the human motion analysis problem is the extraction of moving targets from a video stream. There are three conventional approaches to moving target detection: temporal differencing (two-frame or three-frame) [1], background subtraction [5, 10] and optical flow (see [2] for an excellent discussion). Temporal differencing is very adaptive to dynamic environments, but generally does a poor job of extracting all relevant feature pixels. Background subtraction provides the most complete feature data, but is extremely sensitive to dynamic scene changes due to lighting and extraneous events. Optical flow can be used to detect independently moving targets in the presence of camera motion, however most optical flow computation methods are very complex and are inapplicable to real-time algorithms without specialized hardware.

The approach presented here is similar to that taken in [5] and is an attempt to make background subtraction more robust to environmental dynamism. The notion is to use an adaptive background model to accommodate changes to the background while maintaining the ability to detect independently moving targets.

Consider a stabilized video stream or a stationary video camera viewing a scene. The returned image stream is denoted I_n where n is the frame number. There are four types of image motion which are significant for the purposes of moving target detection: slow dynamic changes to the environment such as slowly changing lighting conditions; "once-off" independently moving false alarms such as tree branches breaking and falling to the ground; moving environmental clutter such as leaves blowing in the wind; and legitimate moving targets.

The first of these issues is dealt with by using a statistical model of the background to provide a mechanism to adapt to slow changes in the environment. For each pixel value p_n in the n^{th} frame, a running average $\overline{p}_n$ and a form of standard deviation σ_{p_n} are maintained by temporal filtering. Due to the filtering process, these statistics change over time reflecting dynamism in the environment.

The filter is of the form

$$F(t) = e^{\frac{t}{\tau}} \quad (1)$$

where τ is a time constant which can be configured to refine the behavior of the system. The filter is implemented:

$$\begin{aligned} \overline{p}_{n+1} &= \alpha p_{n+1} + (1-\alpha)\overline{p}_n \\ \overline{\sigma}_{n+1} &= \alpha |p_{n+1} - \overline{p}_{n+1}| + (1-\alpha)\overline{\sigma}_n \end{aligned} \quad (2)$$

where $\alpha = \tau \times f$, and f is the frame rate. Unlike the models of both [5] and [10], this statistical model incorporates noise measurements to determine foreground pixels, rather than a simple threshold. This idea is inspired by [4].

If a pixel has a value which is more than 2σ from $\overline{p}_n$, then it is considered a foreground pixel. At this point a multiple hypothesis approach is used for determining its behavior. A new set of statistics $(\overline{p}', \sigma')$ is initialized for this pixel and the original set is remembered. If, after time $t = 3\tau$, the pixel value has not returned to its original statistical value, the new statistics are chosen as replacements for the old.

"Moving" pixels are aggregated using a connected component approach so that individual target regions can be extracted. Transient moving objects will cause short term changes to the image stream that will not be included in the background model, but will be continually tracked, whereas more permanent changes will (after 3τ) be absorbed into the background.

3 Target pre-processing

No motion detection algorithm is perfect. There will be spurious pixels detected, holes in moving features, "interlacing" effects from video digitization processes, and other anomalies. Foreground regions are initially filtered for size to remove spurious features, and then the remaining targets are pre-processed before motion analysis is performed.

3.1 Pre-processing

The first pre-processing step is to clean up anomalies in the targets. This is done by a morphological dilation followed by an erosion. This removes any small holes in the target and smoothes out any interlacing anomalies. In this implementation, the target is dilated twice followed by a single erosion. This effectively robustifies small features such as thin arm or leg segements.

After the target has been cleaned, its outline is extracted using a border following algorithm. The process is shown in figure 1.

3.2 "Star" skeletonization

An important cue in determining the internal motion of a moving target is the change in its boundary shape over time and a good way to quantify this is to use skeletonization. There are many standard techniques for skeletonization such as thinning and distance transformation. However, these techniques are

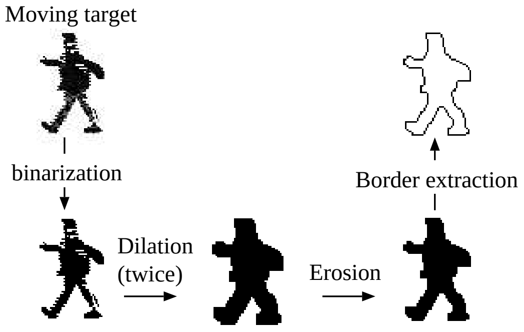


Figure 1: *Target pre-processing. A moving target region is morphologically dilated (twice) then eroded. Then its border is extracted.*

computationally expensive and moreover, are highly susceptible to noise in the target boundary. The method proposed here provides a simple, real-time, robust way of detecting extremal points on the boundary of the target to produce a "star" skeleton. The "star" skeleton consists of only the gross extremities of the target joined to its centroid in a "star" fashion.

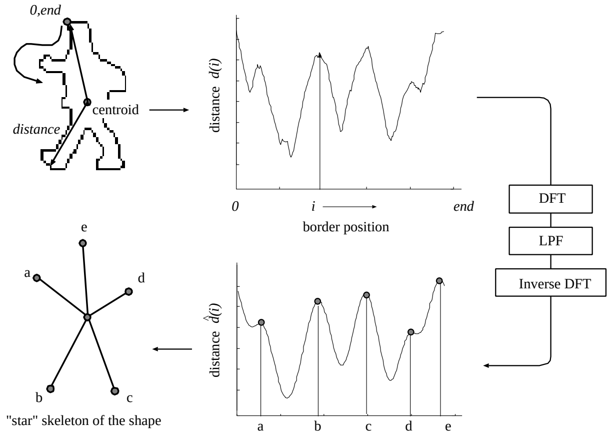


Figure 2: *The boundary is "unwrapped" as a distance function from the centroid. This function is then smoothed and extremal points are extracted.*

1. The centroid of the target image boundary (x_c, y_c) is determined.

$$\begin{aligned} x_c &= \frac{1}{N_b} \sum_{i=1}^{N_b} x_i \\ y_c &= \frac{1}{N_b} \sum_{i=1}^{N_b} y_i \end{aligned} \quad (3)$$

where (x_c, y_c) is the *average* boundary pixel position, N_b is the number of boundary pixels, and (x_i, y_i) is a pixel on the boundary of the target.

2. The distances d_i from the centroid (x_c, y_c) to each border point (x_i, y_i) are calculated

$$d_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} \quad (4)$$

These are expressed as a one dimensional discrete function $d(i) = d_i$. Note that this function is periodic with period N_b.

3. The signal $d(i)$ is then smoothed for noise reduction, becoming $\hat{d}(i)$. This can be done using a linear smoothing filter or low pass filtering in the Fourier domain.

4. Local maxima of $\hat{d}(i)$ are taken as extremal points, and the "star" skeleton is constructed by connecting them to the target centroid (x_c, y_c). Local maxima are detected by finding zero-crossings of the difference function

$$\delta(i) = \hat{d}(i) - \hat{d}(i-1) \tag{5}$$

This procedure for producing "star" skeletons is illustrated in figure 2.

3.3 Advantages of "star" skeletonization

There are three main advantages of this type of skeletonization process. It is not iterative and is, therefore, computationally cheap. It also explicitly provides a mechanism for controlling scale sensitivity. Finally, it relies on no *a priori* human model.

The scale of features which can be detected is directly configurable by changing the cutoff frequency c of the low-pass filter. Figure 3 shows two smoothed versions of $d(i)$ for different values of c: $c = 0.01 \times N_b$ and $c = 0.025 \times N_b$. For the higher value of c, more detail is included in the "star" skeleton because more of the smaller boundary features are retained in $\hat{d}(i)$. So the method can be scaled for different levels of target complexity.

An interesting application of this scalability is the ability to measure the complexity of a target by examining the number of extremal points extracted as a function of smoothing.

Other analysis techniques [10, 6, 5], require *a priori* models of humans – such as the *cardboard model* in order to analyze human activities. Using the skeletonization approach, no such models are required, so the method can be applied to other objects like animals and vehicles (see Figure 4). It is clear that the structure and rigidity of the skeleton are important cues in analysing different types of targets. However, in this implementation, only human motion is considered. Also, unlike other methods which require the tracking of specific features, this method uses only the object's boundary so there is no requirement for a large number of "pixels on target".

4 Human motion analysis

One technique often used to analyze the motion or gait of an individual target is the cyclic motion of skeletal components [9]. However, in this implementation, the knowledge of individual joint positions cannot be determined in real-time. So a more fundamental cyclic analysis must be performed.

Another cue to the gait of the target is its posture. Using only a metric based on the "star" skeleton, it is possible to determine the posture of a moving human.

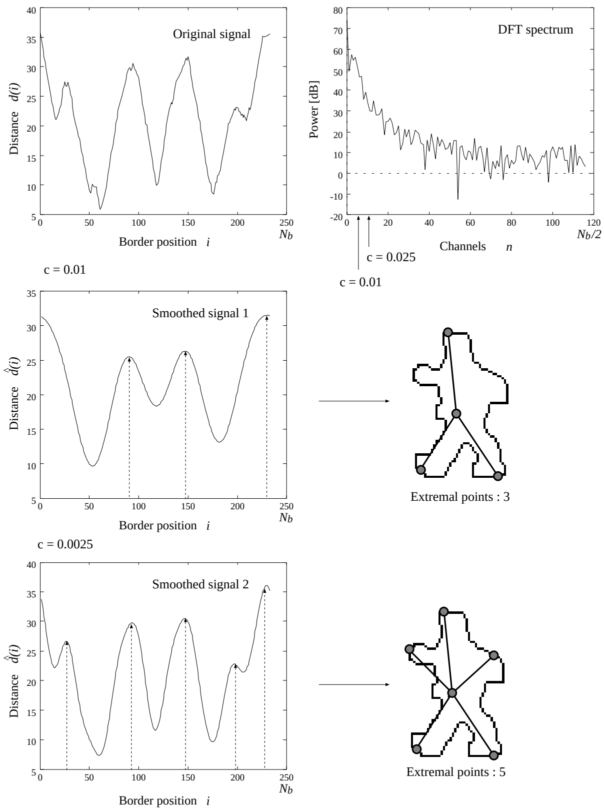


Figure 3: *Effect of cut-off value c. When c is small only gross features are extracted, but larger values of c detect more extremal points.*

4.1 Significant features of the "star" skeleton

For the cases in which a human is moving in an upright position, it can be assumed that the lower extremal points are legs, so choosing these as points to analyze cyclic motion seems a reasonable approach. In particular, the left-most lower extremal point (l_x, l_y) is used as the cyclic point. Note that this choice does not guarantee that the analysis is being performed on the same physical leg at all times, but the cyclic structure of the motion will still be evident from this point's motion. If $\{(x_i^s, y_i^s)\}$ is the set of extremal points, (l_x, l_y) is chosen according to the following condition:

$$(l_x, l_y) = (x_i^s, y_i^s) \; : \; x_i^s = \min_{y_i^s < y_c} x_i^s \tag{6}$$

Then, the angle (l_x, l_y) makes with the vertical θ is calculated as

$$\theta = tan^{-1} \frac{l_x - x_c}{l_y - y_c} \tag{7}$$

Figure 5(a) shows the definition of (l_x, l_y) and θ.

One cue to determining the posture of a moving human is the inclination of the torso. This can be approximated by the angle of the upper-most extremal

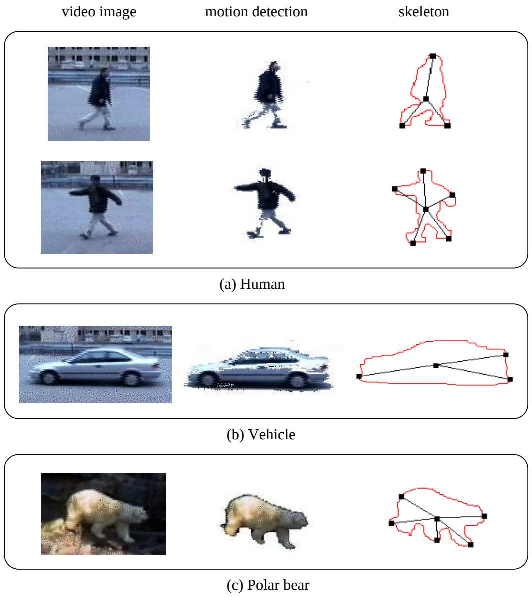


Figure 4: *Skeletonization of different moving targets. It is clear the structure and rigidity of the skeleton is significant in analyzing target motion.*

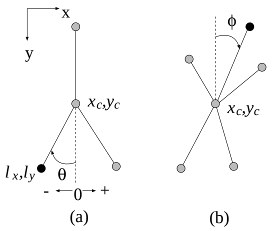


Figure 5: *Determination of skeleton features. (a) θ is the angle the left cyclic point (leg) makes with the vertical, and (b) ϕ is the angle the torso makes with the vertical.*

point of the target. This angle ϕ can be determined in exactly the same manner as θ. See figure 5(b).

Figure 6 shows human target skeleton motion sequences for walking and running and the values of θ_n for the cyclic point. These data were acquired in real-time from a video stream with frame rate 8Hz. This value is not a constant in this technique but depends on the amount of processing which is required to perform motion analysis and target pre-processing.

Note that in figure 6(c), there is an offset in the value of θ_n in the negative direction. This is because only the leftmost leg (from a visual point of view) is used in the calculation and the calculation of θ is therefore biased towards the negative. There is also a bias introduced by the gait of the person. If s/he is running, the body tends to lean forward, and the values of θ_n tend to reflect this overall posture. Another feature which can clearly be observed is that the frequency of the cyclic motion point is clearly higher

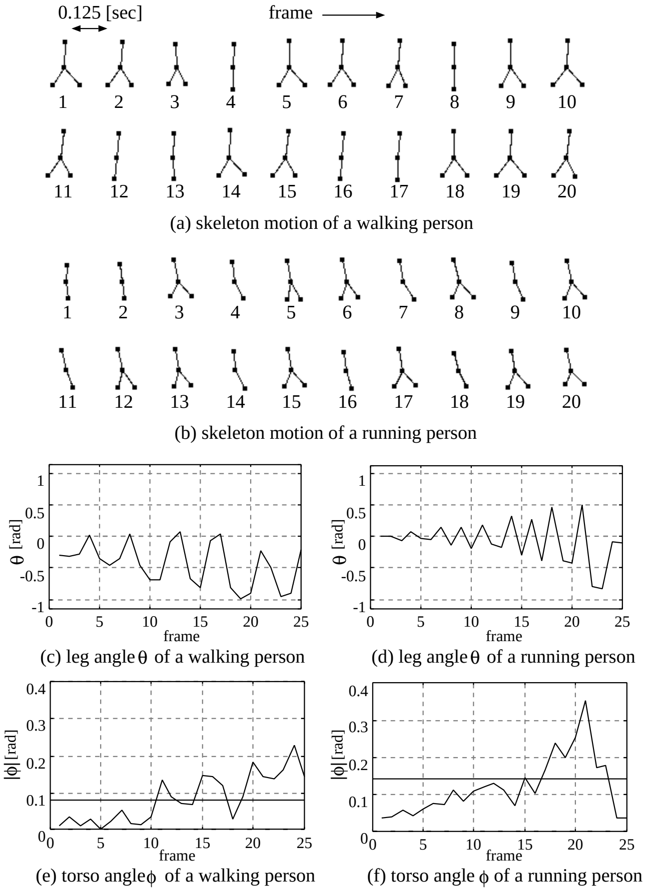


Figure 6: *Skeleton motion sequences. Clearly, the periodic motion of θ_n provides cues to the target's motion as does the mean value of $\overline{\phi}_n$.*

in the case of the running person, so this can be used as a good metric for classifying the speed of human motion.

Comparing the average values $\overline{\phi}_n$ in figures 6(e)-(f) show that the posture of a running target can easily be distinguished from that of a walking one using the angle of the torso segment as a guide.

4.1.1 Cycle detection

Figures 6(c)-(d) display a clear cyclical nature in θ_n. To quantify these signals, it is useful to move into the Fourier domain. However, there is a great deal of signal noise, so a naive Fourier transform will not yield useful results - see figure 7(b). Here, the power spectrum of θ_n shows a great deal of background noise.

To emphasize the major cyclic component, an auto-correlation is performed on θ_n providing a new signal R_i.

$$R_i = \frac{1}{N+1-i} \sum_{n=1}^{N} \theta_n \theta_{n-i} \qquad (8)$$

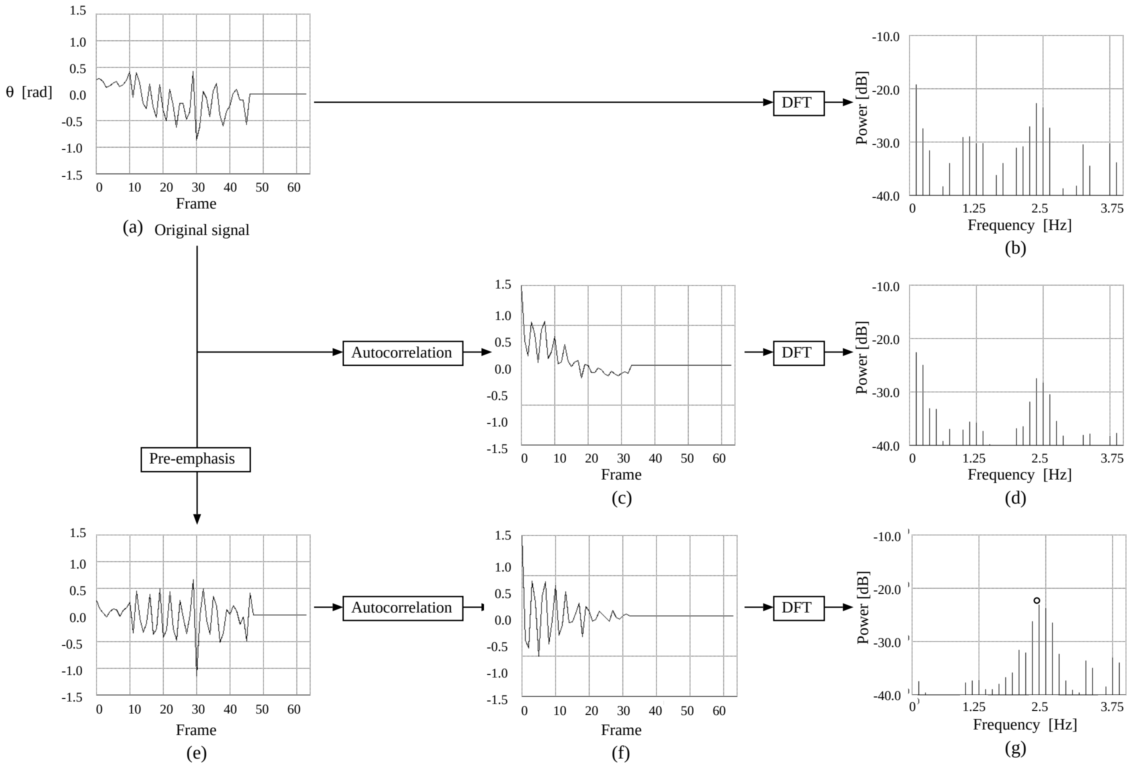


Figure 7: *Process for detecting cyclic motion.*

where N is number of frames. This is shown in figure 7(c).

This autocorrelation process introduces a new source of noise due to the bias (or DC component) of the θ_n signal. When low frequency components are autocorrelated, they remain in the signal and show up in the power spectrum as a large peak in the low frequencies with a degeneration of 6 [dB/oct] in the case of figure 7(d). To alleviate this problem, a high frequency pre-emphasis filter $H(z)$ is applied to the signal before autocorrelation. The filter used is:

$$H(z) = 1 - az^{-1} \tag{9}$$

with a chosen empirically to be ≈ 1.0. This yields the figure shown in figure 7(e).

Finally, figure 7(g) shows that the major cyclic component of the cyclic point can be easily extracted from the power spectrum of this processed signal.

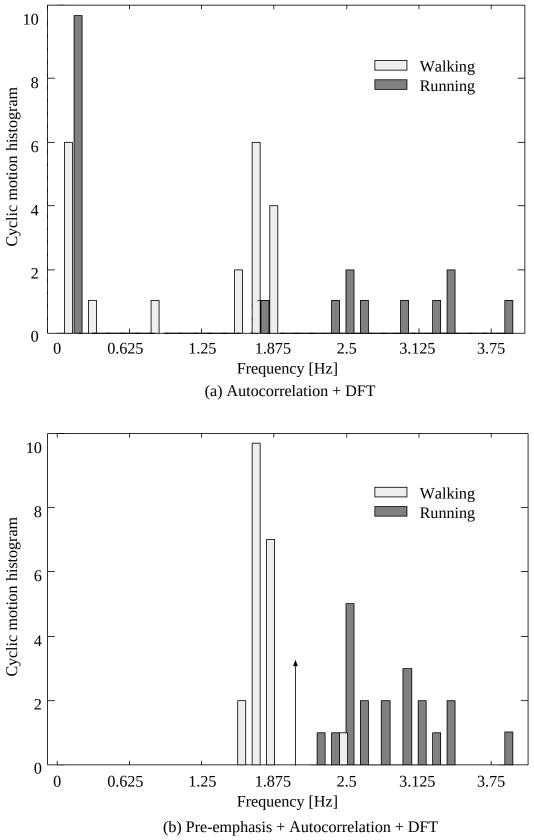
5 Analysis

This motion analysis scheme has been tried on a database of video sequences of people walking and running. There are approximately 20 video sequences in each category, with pixles on target ranging from ≈ 50 to ≈ 400. The targets are a mixture of adults and children. The end-to-end process of MTD, target pre-processing, and motion analysis was performed on an SGI O2 machine containing an R10000 175Mhz processor.

Figure 8 shows histograms of the peaks of the power spectrum for each of the video streams. It is clear from figure 8(a) that the low frequency noise would cause a serious bias if motion classification were attempted. However, figure 8(b) shows how effective the pre-emphasis filter is in removing this noise. It also shows how it is possible to classify motion in terms of walking or running based on the frequency of the cyclic motion. The average walking frequency is 1.75[Hz] and for running it is 2.875[Hz]. A Threshold frequency of 2.0[Hz] correctly classifies 97.5% of the target motions. Note that these frequencies are twice the actual footstep frequency because only the visually leftmost leg is considered. Another point of interest is that the variance of running frequencies is greater than that of walking frequencies, so it could be possible to classify different "types" of running such as jogging or sprinting. For each video sequence, the average inclination $\overline{\phi}$ of the upper extremal point (or torso) was determined. These values are shown in figure 9. It can be seen that the forward leaning of a running figure can be clearly distinguished from the more vertical posture of a walking one. A threshold value of 0.15[rads] correctly classifies 90% of the target motions.

6 Conclusion

Analyzing human motion for video applications is a complex problem. Real-world implementations will have to be computationally inexpensive and be ap-



(a) Autocorrelation + DFT

(b) Pre-emphasis + Autocorrelation + DFT

Figure 8: *Histogram of cyclic motion frequency peaks. (a) The bias in θ_n often produces a frequency peak which is significantly higher than the peak produced by cyclic motion. (b) The pre-emphasis filter effectively removes this noise.*

plicable to real scenes in which targets are small and data is noisy. The notion of using a target's boundary to analyze its motion is a useful one under these conditions. Algorithms need only be applied to a small number of pixels and internal target detail, which may be sketchy, becomes less important.

This paper presents the approach of "star" skeletonization by which the component parts of a target with internal motion may easily, if grossly, be extracted. Further, two analysis techniques have been investigated which can broadly classify human motion. Body inclination can be measured from the "star" skeleton to determine the posture of the human, which derives clues as to the type of motion being executed. In addition, cyclic analysis of extremal points provides a very clean way of broadly distinguishing human motion in terms of walking and running and potentially even different types of gait.

In the future, this analysis technique will be applied to more complex human motions such as crawling, jumping, and so on. It may even be applied to the gaits of animals.

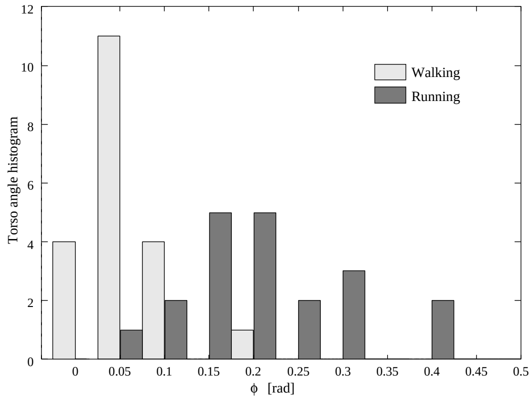


Figure 9: *Average inclination histogram of torso for classification.*

References

[1] C. Anderson, P. Burt, G. van der Wal "Change detection and tracking using pyramid transformation techniques" *SPIE - Intelligent Robots and Computer Vision* Vol. 579, pp. 72-78, 1985

[2] J. Barron, D. Fleet, S. Beauchemin "Performance of Optical Flow Techniques" *International Journal of Computer Vision*, Vol. 12, no. 1, pp. 42-77, Jan. 1994

[3] J. Davis, A. Bobick "The representation and recognition of human movement using temporal templates" *Proceedings of IEEE CVPR 97*, pp. 928 - 934, 1997

[4] E. Grimson, P. Viola "A Forest of Sensors" *DARPA - VSAM workshop*, Nov. 1997

[5] I. Haritaoglu, D. Harwood, L. S. Davis "W^4 Who? When? Where? What? A Real Time System for Detecting and Tracking People" *FGR98 submitted*, 1998

[6] S. Ju, M. Black, Y. Yacoob "Cardboard People: A Parameterized Model of Articulated Image Motion" *International Conference on Face and Gesture Analysis*, 1996

[7] T. Kanade, R. Collins, A. Lipton, P. Anandan, P. Burt "Cooperative Multisensor Video Surveillance" *Proceedings of DARPA Image Understanding Workshop 1997*, Vol. I, pp. 3-10, 1997.

[8] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, T. Poggio "Pedestrian detection using wavelet templates" *Proceedings of IEEE CVPR 97*, pp. 193-199, 1997

[9] P. Tsai, M. Shah, K. Ketter, T. Kasparis "Cyclic motion detection for motion based recognition" *Pattern Recognition*, Vol. 27, No. 12, pp. 1591-1603, 1994

[10] C. Wren, A. Azarbayejani, T. Darrell, A. Pentland "Pfinder: Real-Time Tracking of the Human Body" *IEEE Transactions on Pattern Analysis and Machine Intelligence* July 1997, vol 19, no 7, pp. 780-785