A Proposal to Exploit Contour Cues for Biological Motion Processing

Christoph Rasche
Institut für Psychologie
Justus-Liebig Universität, 35394 Giessen, Germany
Email: rasche15@gmail.com

***Abstract*— We introduce a method to describe the geometry of contours, which is potentially benefical for biological motion perception. The method consists of systematic labeling of contour segments for different window sizes, which allows to determine high-curvature points and to derive a parametric description of contours. The method is considered as complementary to existing approaches of biological motion perception, with the particular advantage of providing detailed structural cues, that are useful for rapid detection and classification of biological motion in noisy gray-scale scenes.**

I. Introduction

Biological motion describes the movement of animals or humans - or still-images thereof -, for instance the dynamics of a walking person or the (static) pose of an athlete. There exist already many algorithms which can reasonably characterize biological motion. Two types are particularly useful, algorithms which determine the motion flow field and algorithms using the contour silhouette (see e.g. [1] for a review). Both have their advantages and disadvantages but can complement each other when one fails to perceive motion in certain circumstances [2]. Here, we sketch an approach which accurately describes contour geometry and which is therefore beneficial for, the detection of typical person silhouettes, as well as the detection of silhouette corners to estimate motion speed.

II. Model

The biggest challenge of contour description is the detection of segments which represent object-characteristic features. To segregate meaningful segments, Fischler and Bolles had already proposed a contour-labeling method [3]. We use a very similar labeling method, but extend it to derive parameters reflecting the contour geometry such its degree of curvature, edginess, symmetry and so on.

1) Labeling, Signatures, Local/Global Space: A first step toward the description of contour geometry is to iterate the contour with a window that labels each selected segment. Given a contour with arc length variable v, a chord (or window or segment) of fixed length ω is selected and its endpoints connected by a straight line ℓ. The maximal deviation (or amplitude) a_{max} between the selected segment v_ω and the straight line ℓ is determined. If the segment lies primarily on one side of the straight line ℓ, the segment is labeled a bow and the amplitude a_{max} is assigned to a 'bowness' signature $\beta(v)$ - otherwise the amplitude value is 0. If the segment lies on both sides, it contains a critical point and the segment is labeled an inflexion and the amplitude a_{max} is assigned to an inflexion signature $\tau(v)$. Iterating this labeling process through the entire contour creates the signatures $\beta(v)$ and $\tau(v)$. The signatures are created for a range of window sizes, $\omega \in [1, l_c]$ (l_c = arc length), leading to the local/global space also called *LG space* hereafter:

$$\beta_\omega(v), \quad \tau_\omega(v), \quad \omega \in [1, l_c], \tag{1}$$

The left column in figure 1 shows the LG space for a wiggly arc: at a local level (for small window sizes), the contour appears noisy and the bowness and transition signature appear irregular (e.g. see window no. 3 and 4). At a global level (large window size) the contour appears as an arc, which is expressed by the presence of a single function block for the bowness signature (e.g window no. 6 to 7). The term *function block* describes a range of neighboring signature values which are above 0, e.g. for window no. 5, the first function block for the bowness signature extends from pixels 8 to 15.

A bowness-function block is then characterized more exhaustively. Its degree of circularity $\zeta^\sqcap$ is given by the integral of the function block $\beta^\sqcap$: $\zeta^\sqcap = \int \beta^\sqcap$. To distinguish whether the block describes a L feature or arc, we define a parameter edginess $\epsilon^\sqcap$. It is determined by multiplying the derivate of $\beta^\sqcap$ by a normalized, ramp function F^{ramp}, whose width is equal to the block size (with center value equal 0): $\epsilon^\sqcap = \beta^{\sqcap\prime} F^{ramp}$. The edginess value is largest for a L feature, it is 0 for a perfect arc. The block's symmetry $\upsilon^\sqcap$ is determined by integrating the absolute difference between the first block half $\beta^{\sqcap_1}$ and its reversed second block half $\beta^{\overleftarrow{\sqcap_2}}$, which then is normalized ($l_\sqcap$=block size): $\upsilon^\sqcap = \frac{1}{2l_\sqcap} \int^{l_\sqcap/2} |\beta^{\sqcap_1} - \beta^{\overleftarrow{\sqcap_2}}|$. A value of 0 means complete symmetry, an increasing value corresponds to increasing asymmetry.

2) Contour Boundary Segregation: If a contour contains an 'end' - a turn of 180 degrees - it is segregated at its point of highest curvature. An end can be detected by analyzing the bowness signature from local to global: whenever its amplitude exceeds the length of a half circle with radius equal to half the window size ($\beta_\omega(s) > \omega\pi/2$), then the location of the maximum amplitude is selected as a potential point of segregation. After application of this rule, any contour appears either as elongated in a coarse sense, which can already be described by the above geometries. But this rule will also segregate a smooth arc, whose arc length is larger than 180

978-1-4244-3786-3/09/$25.00 ©2009 IEEE

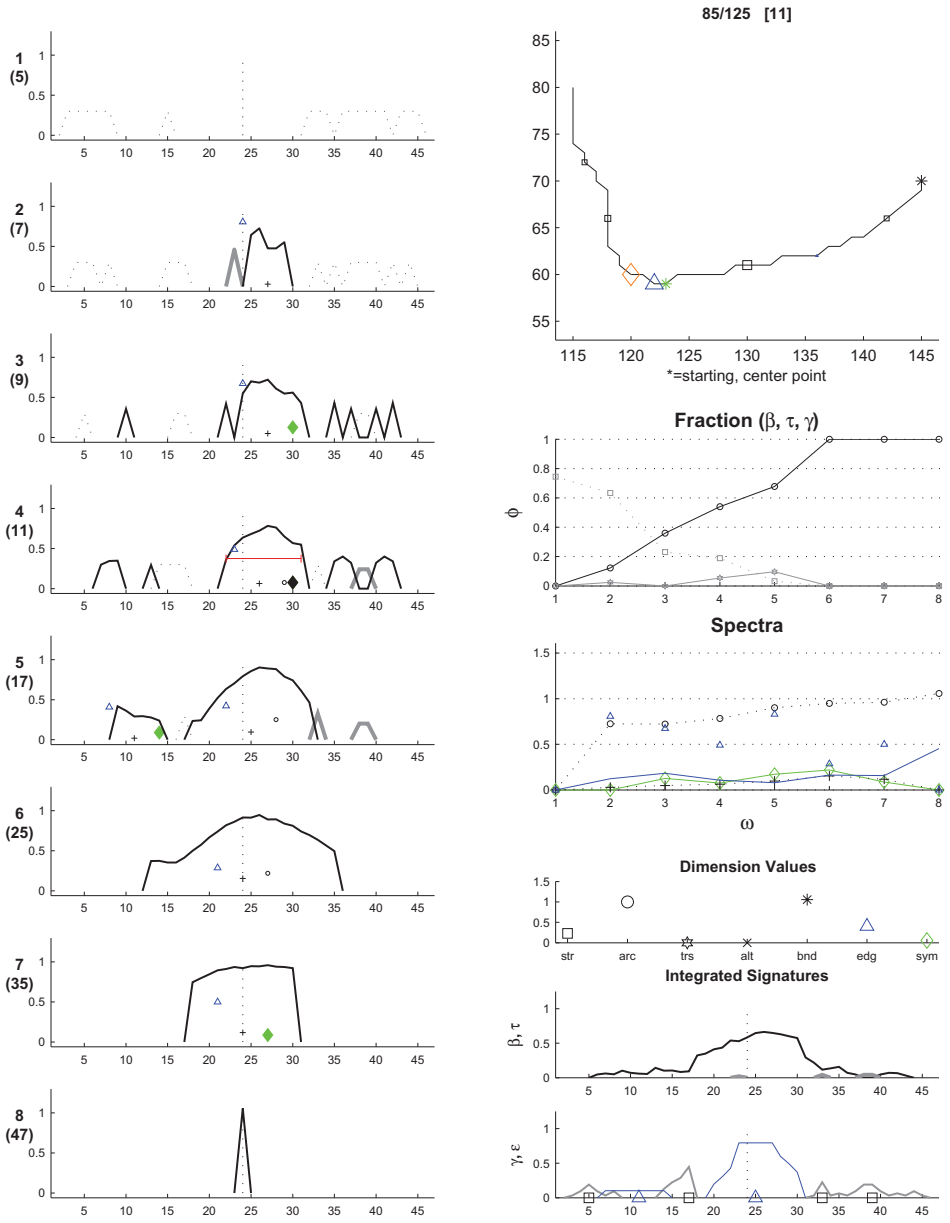

Fig. 1. Local/global (LG) space of a wiggly arc. **Top right**: sample contour with starting and center points marked as asterisk. **Left column**: LG space: signatures $\beta(v)$ (black) and $\tau(v)$ (grey) for 8 different window sizes [x-axis= arc length variable v]. Function block characteristics (determined for large ones only): triangle marker= $\epsilon^{\sqcap}$ (edginess); diamond marker=$v^{\sqcap}$ (symmetry); plus sign marker=$\zeta^{\sqcap}$ (circularity). **Fraction**: fraction ϕ of bowness- and transition-function blocks per window size. **Spectra**: Green diamond: maximum of symmetry value; black circle: maximum β amplitude; plus sign: maximum of ζ. **Dimension Values**: straightness, arc, transition, alternation, bnd=curvature, edginess, symmetry. **Integrated Signatures**: top graph: bowness (black), transition (gray); bottom graph: edginess (triangles) and straightness (squares).

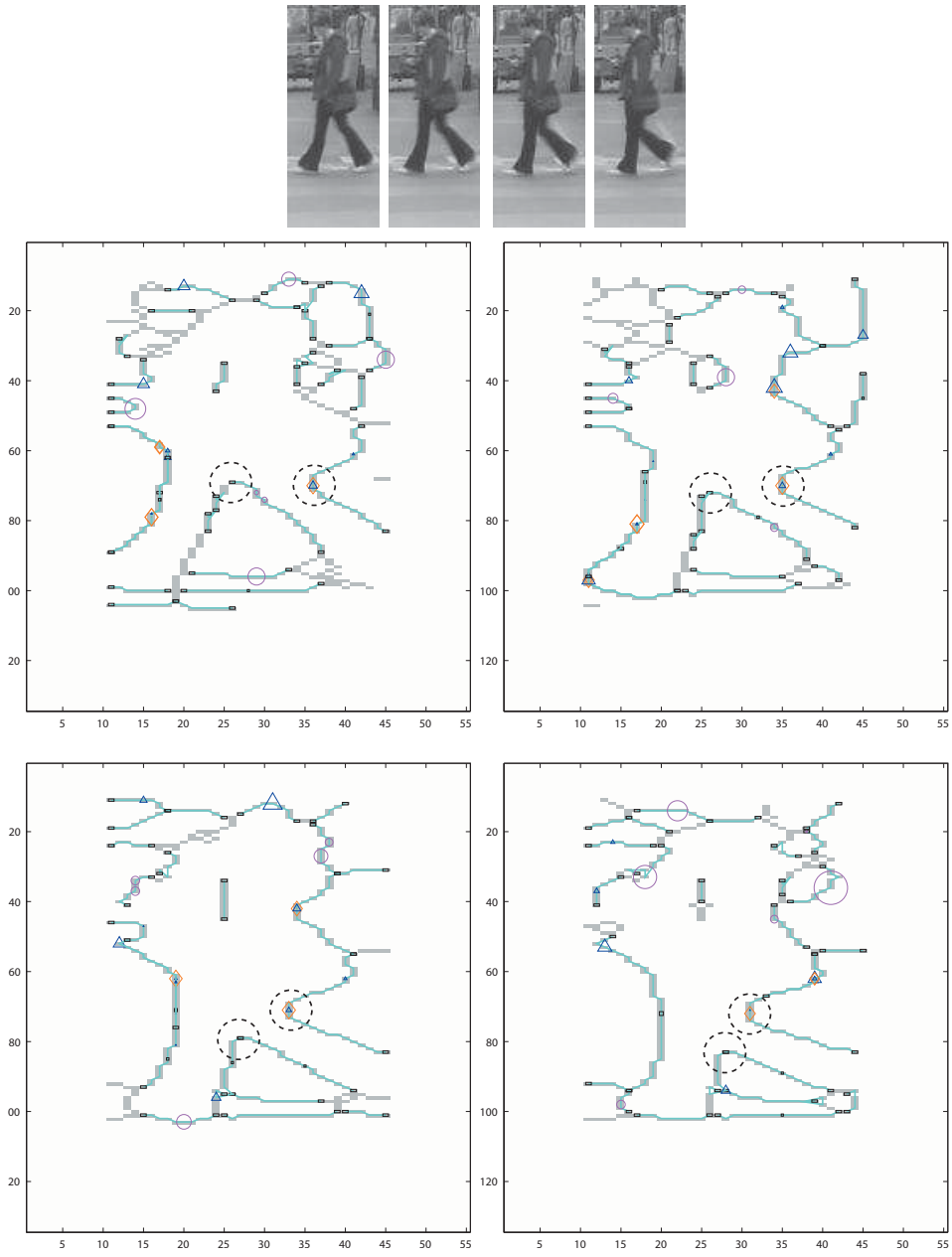

Fig. 2. Labeling output for 4 pedestrian images (row-wise). Contour endpoints marked by small squares; sharp corners marked by triangles; curved contour segments marked by circles; symmetric segments marked by diamonds. Motion cues are given by pursuing the curvatures outlined by the dashed circles.

degrees. Such arcs are extracted and can be detected by analyzing the symmetry of the corresponding function block. Exemplifying these two segregation steps on the Ω shape; the shape is halvened and its circular part is extracted. In a similar way, any irregular contour may contain a 'symmetric' segment which outlines an object part (e.g. smooth arc, L feature or straight line); those segments are extracted as well if they are of a minimum arc length. Thus, the segregation process does not strictly segregate the contours into separate segments, but will create partially overlapping segments to some extent. Taking the Ω shape as the example again, it is segregated into 7 segments: two inflexions representing the halves; one smooth arc segment; two L features representing the corners; and two straight segments (if of sufficient arc length).

3) Parameters for Representation: The LG space can be exploited to extract parameters which describe the contour's global geometry. Three types of global geometries were pursued: arc (L feature or smooth arc), inflexion (change of sign) and alternating (irregularly or evenly), whereby the latter is a sequence of arcs and inflexions. These geomtries can be determined by looking at the fraction of the bowness and inflexion signature as calculated for each window (figure 1,

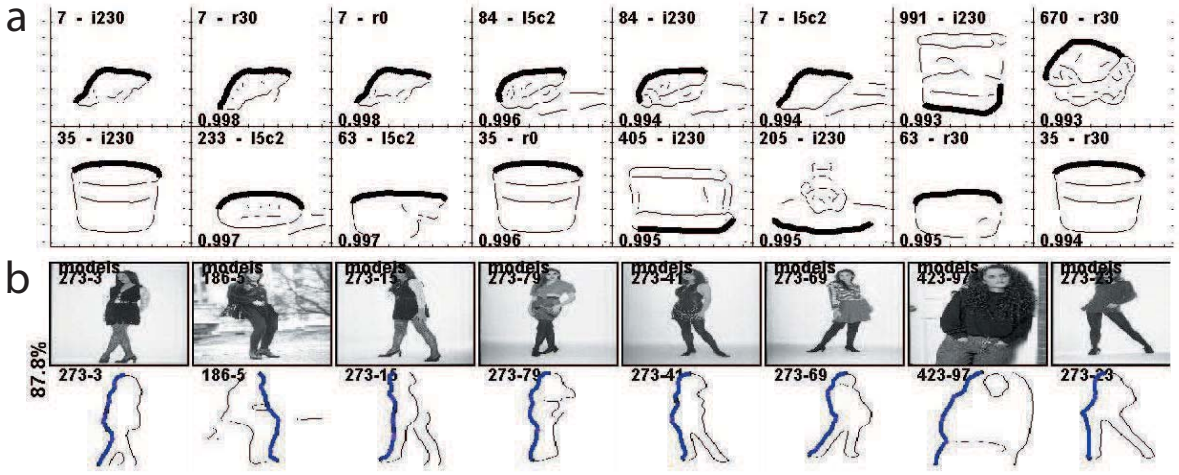

Fig. 3. Similarity-based contour search. The contour of the first image in each row is the selected sample contour, the remaining images in each row contain the most similar contours. **a**. Two examples using the ALOI collection (4000 images, 4 viewpoints for all 1000 objects; >50000 contours). **b**. An example for all contours of the entire Corel collection (60000 images; >1 mio contours). The percentage on the left denotes correct basic-level categorization for the first 99 similar images. For this sorting, 5 appearance parameters were included (13-dimensional vector).

right column 'Fraction'). Other parameters are extracted, such as the curvature of the contour (e.g. the maximum amplitude of the bowness space), the edginess (e.g. the maximum value of all function blocks), the symmetry, and so on. Those parameters are then concatenated to form a multi-dimensional vector with ca. 8 geometric dimensions (see figure 1, right column, 'Dimension Values' for some parameter values).

III. Implementation and Results

Window sizes were generated in increments of $\sqrt{2}$. The smallest window sizes were $\omega = [5, 7, 9, 11, 13]$ (number of contour pixels) for scale $\sigma = 1$. The signatures are normalized by their window lengths (a_{max}/ω) - this is a crude approximation but computationally cheap. For that reason, the threshold for detecting potential ends was set heuristically (value=1.2) as well as the threshold for detecting circularity (value=90). The most time consuming part of our algorithm is the extraction of the contour lists (Matlab algorithm), but the generation of the LG spaces and the derived spectra and parameter values requires much less time, as only computations with 1D signals are performed. The total computation time is a fraction of a second.

Figure 2 shows the contour labeling output on four pedestrian images. Salient silhouette points are either marked as sharp corners (triangles) or were segregated as part of an end (dashed circles). Figures 3a and b demonstrate the specificity of the contour vector by performing a similarity search using the ALOI [4] and COREL collection: A contour was selected (depicted in the first image) and compared to all other contours of all other images and then ordered by decreasing similarity (increasing distance). In the sorting shown in figure 3a similar contours were found in the same object seen under slightly different views; in figure 3b shows that the key feature for persons is a vertical alternating contour with sharp corners.

IV. Conclusion

Figure 3a demonstrated that the multi-dimensional contour vectors are highly specific; figure 3b showed that (static) biological motion can be detected relatively easily from a large image collection. This high contour specificity could be possibly further elaborated and exploited to estimate the pose of humans. Figure 2 showed that the high-curvature points of a silhouette are detected. Those points and the segregated contour segments could provide essential hints for motion description. If those high-curvature points and segments were integrated in an intelligent manner, then those features could serve as pedestrian detectors. The presented method is relatively fast and could therefore be useful for the rapid prediction of the presence of biological motion which is essential for gaze-guidance systems [5].

Acknowledgment

The study is supported by the Gaze-Based Communication Project (European Commission within the Information Society Technologies, contract no. IST-C-033816). The author would like to thank David Engel for providing the pedestrian images.

References

[1] R. Poppe, "Vision-based human motion analysis: An overview." *Computer Vision and Image Understanding*, vol. 108, pp. 4–18, 2007.

[2] N. R. Howe, "Flow lookup and biological motion perception." *IEEE International Conference on Image Processing, September 11-14, 2005, Genoa, Italy*, vol. 3, pp. 1168–1171, 2005.

[3] M. Fischler and R. Bolles, "Perceptual organization and the curve partitioning problem," in *Proceedings of the Tenth International Joint Conference on Artificial Intelligence, Volume 2*, 1983.

[4] J. M. Geusebroek, G. J. Burghouts, and A. W. M. Smeulders, "The Amsterdam library of object images," *Int. J. Comput. Vision*, vol. 61, no. 1, pp. 103–112, 2005.

[5] E. Barth, M. Dorr, M. Bhme, K. Gegenfurtner, and T. Martinetz, "Guiding the mind's eye: improving communication and vision by external control of scanpath," *Human Vision and Electronic Imaging: Proceedings of SPIE*, vol. 6057, pp. 1–8, 2006.