Proceedings of the 5th WSEAS Int. Conf. on Signal Processing, Computational Geometry & Artificial Vision, Malta, September 15-17, 2005 (pp107-111)

A Curve Matching Algorithm for Dynamic Image Sequences

VLADAN PAPIC
Polytechnics department
University of Split
Teslina 12, 21000 Split
CROATIA
http://www.pmfst.hr/~vpapic

HRVOJE DUJMIC
Electronics department
University of Split
Rudjera Boškovica bb, 21000 Split
CROATIA

Abstract: - In this paper, a curve matching algorithm for the detection of moving object has been developed. The complete procedure of curve extraction, registration, tracking and matching is described. Curves are represented using the 3rd order polynomial approach. Searching window in the successive dynamic images of the observed video sequence is narrowed using the motion estimator. In each image, the curve of winner chain is selected by a novel algorithm based on calculating the highest similarity coefficient. The similarity coefficient is the compilation of multiple factors. The proposed algorithm is evaluated through tests that show high reliability in real world problems.

Key-Words: - Curve matching, Similarity coefficient, Dynamic images, Winning chain

1 Introduction

Curve matching, in essence, has the goal of identifying curves in images. It is an important problem in computer vision with many applications such as recognition based on pre-defined models or templates, object tracking, depth from stereo, image database categorization. Usually, it is a part of image segmentation stage but it could be implemented without the broader scope of segmenting the image.

Similar term used in this area of research is curve tracking. Curve tracking includes extracting the curves and identifying them in images [3]. Both stages are equally important because successful curve identification and automatic classification can't be done if the candidate curves are inaccurately extracted or even some of them are not extracted at all. Generally, a person can identify curves with ease and the intention is to understand human recognition process and to implement it for the computer vision. Some complicated multiple mutually intersected curves can be the problem even for the human observer and these problems are consequently harder to resolve using the image processing methods and algorithms.

Matching algorithms are also needed for applications like image database categorization, where the dominant silhouettes are searched in order to classify particular image.

Curve matching algorithms are basically comparing two curves and deciding whether the curves similarity satisfies some predefined criteria. Acceptance threshold depends on image noise level and allowed 2D rigid transformations (for example, expected size variations, expected orientation variations etc.). In the case of curve tracking, matching is performed for all the candidate curves but only one curve can be classified as the curve that corresponds to the model curve.

There should be made a distinction between algorithms used for the closed and open curves. Contour matching algorithms are generally dealing with the closed curves and closed contour can be represented with sequence of connected and ordered open curves.

Finding the particular segment of the entire boundary i.e. starting and ending points of the segment that will be presented as a curve is often the one of the problems that should be resolved [4][5]. Also, intensive research has been done on construction of the smooth curves and the digitalization errors [6].

Most of the work on contour matching involves direct curve matching. The parameterization problem and solving the cost function are the topics of various papers [16], [17].

Feature matching methods may involve different approaches. Search for the best matching while permitting the rotation, translation and scaling of each curve, such that the distances between matched key points are minimized [10],[11]. As an alternative

Proceedings of the 5th WSEAS Int. Conf. on Signal Processing, Computational Geometry & Artificial Vision, Malta, September 15-17, 2005 (pp107-111)

to the alignment transformation, features may be mapped to an intrinsic invariant coordinate frame [12]. Algorithms that seek isomorphism between attributed relational graphs [13], and algorithms that look for the largest set of mutually compatible matches [14], [18].

Syntactical matching methods are inspired by string comparison algorithms, which use substitution, deletion and insertion operations to transform one string to the other [15].

In curve matching applications involving motion detection some additional methods and techniques should be applied. Motion detection of curves deals with relating spatial image sequences to temporal changes [2]. The optical flow [7] is the parameter provided by the sequence of 2D images. Important aspect of motion detection is motion estimation. Motion estimation in dynamic video sequences was investigated by Bregler et al [8] and Kakadiaris et al [9]. It has been the subject of interest due to estimation significance in the recognition and tracking process robustness, accuracy and speed.

Our approach is based on the non-closed, simple curves tracking in the dynamic images sequences. The reason for narrowing the problem space was found in fact that in large number of applications there is no need of tracking or searching for the whole objects presented with their closed edges. Some part of the object silhouette is frequently enough for the recognition of the whole object. Curves are tracked in the sequence of images so additional information for successful recognition of the curve in the image was available due to applied motion estimation.

The basic idea is described in the second section of the paper. Matching algorithm will be explained in the third section, the results of the algorithm performance are given it section four and the conclusion is given in section five.

2 Curve Matching Approach

First, lets start with the general tracking procedure shown in Fig. 1. It consists of two main stages – candidate curves extraction and curve matching. Both of these phases are receiving additional information from database. Also, the output from the curve matching module updates database in order to improve performance for the curve recognition in successive frames. The database holds the information about the original curve used as a model. It contains the application – specific knowledge such as maximal linear and angular velocities of the objects being tracked, maximal curve lengths, heuristically or experimentally determined weights of the factors needed for the winner chain selection as well as other rules and facts that can prove to be helpful in the tracking process. Also, the information about the winner chains positions in the previous frames is recorded.

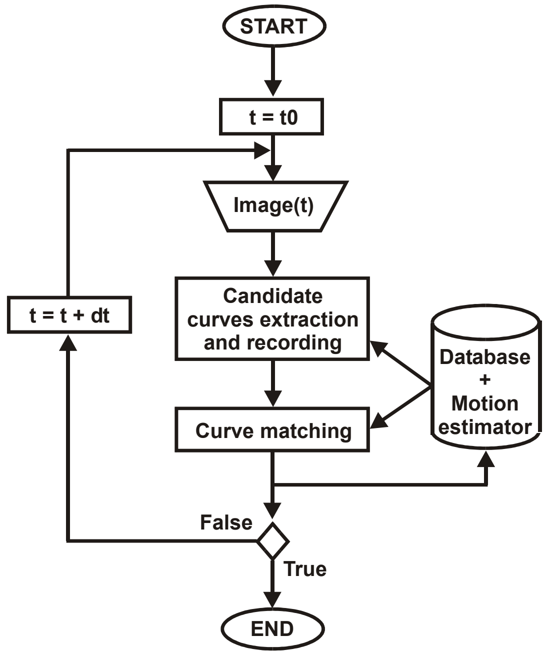

Fig.1 General tracking procedure

Images extracted from the video sequences are shifted in time for dt.

The curves extraction and recording module use line following function. This function uses database informations for determination of starting and ending points of the chains as well as expected curve shapes in order to trigger the right decision in the case of edges bifurcations. Chain coding [1] is used for curve recording (coding).

Motion estimation is very important because a good estimation implies processing speed multiplication and better reliability of the obtained results. Estimated position, as well as the curve orientation, depends on position and orientation in previous images of the sequence. We can write the following equation:

$$S_i\left[\left(x_{j,i}, y_{j,i}, \Theta_{j,i}\right)\right] = \alpha\left\{S_{i-1}\left[\left(x_{j,i-1}, y_{j,i-1}, \Theta_{j,i-1}\right)\right], \left(\Delta_{j,i}\right)\right\}, \quad j = 1,...n \qquad (1)$$

where

$S_i\left[\left(x_{j,i}, y_{j,i}, \Theta_{j,i}\right)\right]$ is set of estimated positions and orientations for j-th curve of the i-th image. Total number of curves is n.

$\Delta_{1,i}, \Delta_{2,i}, ..., \Delta_{n,i}$ are calculated shift values on x and y axes of the i-th image

α - function that gives the estimation of current position and orientation on the basis of previous position and orientation values and calculated curve shifts.

Proceedings of the 5th WSEAS Int. Conf. on Signal Processing, Computational Geometry & Artificial Vision, Malta, September 15-17, 2005 (pp107-111)

2.1 Simplifications

Some simplifications were made in order to make the algorithm fast and simple. First, as we have already mentioned, only open curves were chosen for the tracking. Second, it was expected that the curves can be approximated by third order polynomial. Third, the size of the curve being tracked is not expected to be changed significantly. Fourth, the movement of the object and the curve that corresponds to part of its boundary is expected to be smooth. Fifth, good output of the edge detection phase was expected which means that the edges were not interrupted so no additional curve fitting was needed.

2.2 Flexibility

Although, there are constraints mentioned in the previous section, special attention was paid on flexibility of the curve matching algorithm. In order to achieve this goal, all the weight factors can be easily adopted for various kinds of application.

2.3 Premises

The images used as the input for the presented procedure are pre-processed black and white edge images. This means that RGB images acquired from the sensor device are translated to HSI, filtered and the some kind of edge detector is applied. In our case, we used median filtering along with the Canny edge detection but this combination is not mandatory.

3 Algorithm

Curve matching algorithm implemented in curve matching module in Fig. 1 consists of two submodules: polynomial chain approximation submodule and similarity coefficient calculation submodule.

For the reasons of schematic clarity, the polynomial approximation submodule is presented by two separate shapes although the algorithm for the original chain (template) and every other candidate chain is the same.

Benefits of the polynomial representation of the chain are:

- only four polynomial coefficients along with the starting and ending point are sufficient for the chain registration
- possibility of the chain reconstruction without the resolution constrains of the original image
- avoided image metrics constrains in curve representation and comparison

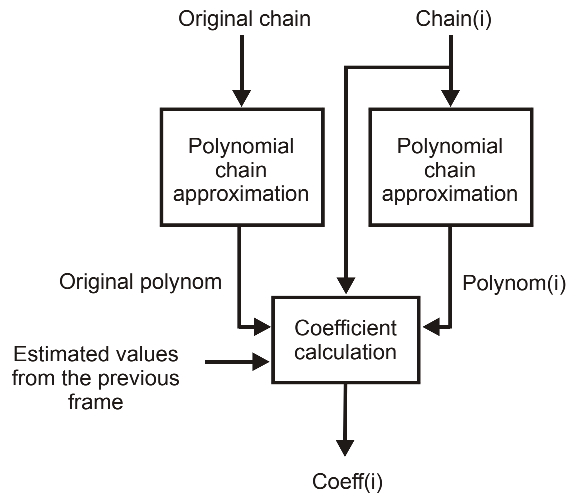

Fig.2 Curve matching: Assigning value to each candidate chain.

Similarity coefficient calculation submodule in Fig.2 should be described in detail. Coefficient of the i-th chain is calculated according to equation:

$$coeff(i) = \begin{bmatrix} c_1 & c_2 & c_3 & c_4 & c_5 \end{bmatrix} \cdot \begin{bmatrix} f_1(x_i, x_{est}) \\ f_2(y_i, y_{est}) \\ f_3(\theta_i, \theta_{est}) \\ f_4(poly(i), poly(1)) \\ f_5(poly(i), poly(1)) \end{bmatrix}$$

(2)

where c_1, c_2, c_3, c_4, c_5 are the weights of the distance measure component functions.

Function f_1 calculates the difference measure of starting x coordinate of i-th chain (x_i) and estimated x coordinate (x_{est}). For the particular case, the square difference of these two coordinates $(x_i - x_{est})^2$ is normalized with the image searching area.

Function f_2 is analogue to f_1 and it calculates the difference measure of starting y coordinate of i-th chain (y_i) and estimated y coordinate (y_{est}).

Function f_3 calculates the difference measure of curve angle of i-th chain (θ_i) and estimated y coordinate (θ_{est}). This function is normalized with maximal tolerated curve angle change between successive frames.

Function f_4 calculates the correlation coefficient between polynomial representations of original chain and current chain.

Function f_5 calculates the standard deviation between polynomial representations of original chain and current chain.

For example, weights and distance measuring functions can be chosen as in

Proceedings of the 5th WSEAS Int. Conf. on Signal Processing, Computational Geometry & Artificial Vision, Malta, September 15-17, 2005 (pp107-111)

$$
\begin{aligned}
c_1 \cdot f_1 &= (2 \cdot (x_i - x_{est}))^2 \\
c_2 \cdot f_2 &= (4 \cdot (y_i - y_{est}))^2 \\
c_3 \cdot f_3 &= (\theta_i - \theta_{previous\_frame})^2 \\
c_4 \cdot f_4 &= 15 \cdot (1 - corr.coeff(i)) \\
c_5 \cdot f_5 &= 10 \cdot ssd(original.poly, poly_i)
\end{aligned} \quad (3)
$$

where ssd is the standard deviation function. Weight matrix $\begin{bmatrix} 2 & 4 & 1 & 15 & 10 \end{bmatrix}^T$ is determined according to experimental results performance and heuristically estimated values.

After the calculation for each chain candidate, chain with the heights score becomes the winner chain. Calculation is repeated for all the frames of the sequence.

4 Results

Software implementation of the algorithm and the whole procedure was done using the Matlab. The proposed algorithm as a part of the general tracking procedure was tested on the dynamic sequences of human walking. For the gait analysis application, one of the important features was the determining the foot – floor angle.

Fig. 3 Images from the test sequence. Tracked curve is presented with white color.

From the first image of the walking sequence, the model curve was taken. This curve for one of the test sequences is presented with the white color in the leftmost subfigure of the Fig. 3.

The foot angle calculated from the curve tracked for all the images of particular sequence showed high accuracy. Eleven sequences with total of 358 images were processed and the curves were successfully detected in 354 images (98.9%). After visual observation of the four problematic images, the reliability of the algorithm was proven even more – foot edge curve was not detectable because of lighting problems, even for humans.

The processor time consumed by the algorithm was insignificant comparing to pre-processing time (RGB to HSI and edge detection).

5 Conclusion

In this paper, a curve matching algorithm for the detection of moving object has been described. We have shown that the proposed algorithm has high reliability in real world problems. The algorithm is flexible because weights of the component factors can be changed according to the specific application demands.

A curve tracking in dynamic images sequences allows simplifications that are used by the presented algorithm. Used simplifications (size invariability, possible position and orientation estimation, acceptable 3rd order polynomial approximation) allowed us to develop an algorithm that is fast enough for the real time applications.

References:

[1] H. Freeman, Computer Processing of Line-Drawing Images, *ACM Comput. Surveys*, Vol.6, No.1, 1974, pp. 57-98.

[2] P. Fua, A. Gruen, N. D'Apuzzo, and R. Plaenkers, Markerless full body shape and motion capture from video sequences, *International Archives of Photogrammetry and Remote Sensing*, Vol.34, B4, 2002, pp. 256-261.

[3] K. Raghupathy, T. Parks, Improved Curve tracing in images, IEEE International Conference on Signal Processing, May, 2004.

[4] T. Davis, Fast Decomposition of Digital Curves into Polygons using the Haar Transform, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 8., 1999.

[5] N. Katzir, M. Lindenbaum, and M. Porat, Curve Segmentation under Partial Oclusion, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 5., 1994.

[6] M. Worring, A. Smeulders, Digitized Circular Arcs: Characterization and Parameter Estimation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 6., 1995.

[7] Bernd Jahne, Horst Haubeker, *Computer Vision and Applications*, Academic Press, San Diego, 2000., pp. 347-395.

[8] C. Bregler, J. Malik, K. Pullen, Twist Based Acquisition and Tracking of Animal and Human Kinematics", *International Journal of Computer Vision*, 56(3), 2004., pp. 179-194.

[9] I. Kakadiaris, D. Metaxis, Model – Based Estimation of 3D Human Motion, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 12., 2000., p.p. 1453-1459.

[10]N. Ayach, O. Faugeras, HYPER: a new approach for the recognition and positioning of two dimensional objects, *Transactions on Pattern*

Analysis and Machine Intelligence , Vol. 8, 1986., pp. 44-54.

[11]B. Kamger-Parsi, M. Margalit, A. Rozenfeld, Matching general polygonal arcs, *CVGIP: image understanding 53*, 1991., pp. 227-234.

[12]S. Sclaroff, A. Pentland, Modal matching for correspondence and recognition, *Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, 1995., pp. 545-561.

[13] S. Li., Matching: invariant to translations, rotations and scale changes, Pattern Recognition, Vol. 25, 1992., pp. 583-594.

[14]Y. Lamdan, J. Schwartz, H. Wolfson, Affine invariant model based object recognition, *IEEE Transactions on Robotics and Automation*, Vol. 6, 1990., pp. 578-589.

[15] Y. Wang, T. Pavlidis, Optimal correspondence of string subsequences, *Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, 1990., pp. 1080-1086.

[16] R. Basri, L. Costa, D. Geiger, and D. Jacobs, Determining the similarity of deformable shapes, *IEEE Workshop on Physics Based Modeling in Computer Vision*, 1995, pp. 135-143.

[17]I. Cohen, N. Ayache, and P. Sulger, Tracking points on deformable objects using curvature information, *Proceedings of ECCV*, 1992., pp. 458-466.

[18] D. Ballard, Generalizing The Hough Transform To Detect Arbitrary Shapes, *Pattern Recognition*, Volume 13, Number 2, 1981., pp. 111-122.