Contour Based Superquadric Tracking

Jaka Krivic and Franc Solina

Computer Vision Laboratory, Faculty of Computer and Information Science
University of Ljubljana
Tržaška 25, 1000 Ljubljana, Slovenia
{jaka.krivic,franc.solina}@fri.uni-lj.si
http://lrv.fri.uni-lj.si

Abstract. This paper proposes a technique for tracking a superquadric-modelled object over a monocular video sequences. The object is currently modelled with a single superquadric. Object's position and orientation in the first frame of the sequence are assumed known. A frame in a sequence is first processed to find object's contour. Contour is determined by extracting edges on the frame in the vicinity of model's contour from the previous frame. The model's relative translation and rotation parameters are then calculated by fitting model's contour to the frame's contour. This fitting is achieved by minimizing the cost function, which is based on model to image mapping.

Keywords: Superquadrics, object tracking, contours

1 Introduction and Motivation

Object tracking from video sequences is a well researched area of computer vision, with a wide variety of possible applications. Common to these applications is that they need to extract information about object's motion from image sequences. The *model based* approach to object tracking assumes that the object is somehow modelled, and the goal is then to match the model to the image. One of the more active fields of object tracking research is tracking of 3D articulated objects, especially human body, using 3D volumetric models [2, 3, 5, 6, 7, 8]. Tracking the object, while it is moving and articulating in front of the camera is done by fitting some kind of projection of the 3D model to the image.

Superquadrics are a common building block for modelling articulated objects [1]. One of the more appreciated properties is their very compact representation, while on the other hand the failure to perfectly describe many natural shapes (especially the concave ones) is their main disadvantage [1]. Nevertheless, in 3D human modelling they are quite frequently used [3, 7, 8, 4]. Although the models are not very photorealistic, they are very appropriate for high-level reasoning.

Many human tracking methods use superquadrics for object modelling [3, 7, 8, 9]. In the approaches of Sminchisescu [7, 8], the superquadrics are used in for modelling only and are then turned to parameterized meshes. In our opinion,

V. Palade, R.J. Howlett, and L.C. Jain (Eds.): KES 2003, LNAI 2774, pp. 1180–1186, 2003.

this is an unnecessary step, especially in [7]. It is not obvious what do the authors gain with discretization of superquadric models. The method we propose does not use any intermediate representations, but tries to fit superquadric models directly. We believe that this reduces computational cost, thus leading to real-time 3D object tracking. Although off-line body tracking has its applications (e.g. virtual character animation, sports movement analysis), it is not useful for human-computer interaction. We are interested in realtime object tracking with relation to human action comprehension.

Next section of this paper first describes the superquadric models used and their properties used in the cost function design, then the method is presented in section 3 and some experimental results in section 4. The paper is concluded with the outline of the future work.

2 Superquadric and Its Contour

Superquadrics are mathematical solids, whose surface is defined by the following implicit equation:

$$\boldsymbol{s}(\eta,\omega) = \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix} = \begin{bmatrix} a_1 \cos^{\epsilon_1}\eta \cos^{\epsilon_2}\omega \\ a_2 \cos^{\epsilon_1}\eta \sin^{\epsilon_2}\omega \\ a_3 \sin^{\epsilon_1}\eta \end{bmatrix}, \qquad \begin{array}{rcl} -\frac{\pi}{2} \leq & \eta & \leq \frac{\pi}{2} \\ -\pi \leq & \omega & < \pi \end{array} \tag{1}$$

The contour of a superquadric is a closed space curve which partitions the superquadric into a visible and invisible part [1].

Observing that the normals of contour points are perpendicular to the camera viewing direction, the analytical form of a contour is

$$\eta(\omega) = \arctan\left[\left(-\frac{a_3}{v_z}\left(\frac{v_x}{a_1}\cos^{2-\epsilon_2}\omega + \frac{v_y}{a_2}\sin^{2-\epsilon_2}\omega\right)\right)^{\frac{1}{2-\epsilon_1}}\right], \qquad -\pi \leq \omega < \pi \tag{2}$$

where $\boldsymbol{v} = [v_x, v_y, v_z]^T$ is a camera viewing vector rotated to the superquadric's local coordinate frame. When viewing direction is perpendicular (or, in practice, almost perpendicular) to the superquadric's z-axis, the contour is a slice of the superquadric, cut by a plane that is perpendicular to the xy-plane. The contour's ω parameter is therefore a constant, signifying the angle of the slice-plane to the x-axis. In this case, equation

$$\omega = \arctan\left[\left(-\frac{v_x a_2}{a_1 v_y}\right)^{\frac{1}{2-\epsilon_1}}\right], \qquad -\pi \leq \eta < \pi \tag{3}$$

can be used instead.

Approximate sampled contour with equidistantly sampled points is achieved by first sampling N points on the contour, i.e. by computing points on the superquadric's contour $T'_i = P(s(\omega_i, \eta(\omega_i)))$, using equation 2, where $\omega_i = 2\pi\frac{i-1}{N} - \pi$, $i = 1 \ldots N$, or $T'_i = s(\omega, \eta_i)$, using equation 3, where $\eta_i = 2\pi\frac{i-1}{N} - \pi$,

$i = 1 \ldots N$. $P(\boldsymbol{x})$ is perspective image projection. Simultaneously contour circumference $C = \sum_{i=1}^{N} d(T'_i, T'_{i-1})$, where $T'_0 = T'_N$ is computed. The rim equidistant points $T_i, i = 1..N$ are then sampled on the lines connecting the successive original sampled contour points, where the distance between successive points is $\frac{C}{N}$.

3 Object Tracking

In the present system the relative translation and rotation of the object from frame to frame are determined based solely on contour information from monocular video. Other information such as additional features or optical flow could be used to improve the tracking results, but the aim of this work was to study the quality of pose estimation from contour information only. In this paper frame contour stands for the object's contour extracted from current frame, while model contour stands for the contour produced by the model.

The first step in processing a frame is frame contour extraction. We did not focus at this step, but rather used a very simple procedure. Frame contour is searched for in the vicinity of the last estimated model contour, i.e. model contour on the previous frame. Maximum relative object motion between successive frames, which is preset, determines the vicinity. In the present system the object contour comprises of the edge features, which give the maximum response in the direction perpendicular to the model contour tangent at the nearest point on model's contour. A more robust method would include model-based silhouette extraction and used the silhouette contour instead.

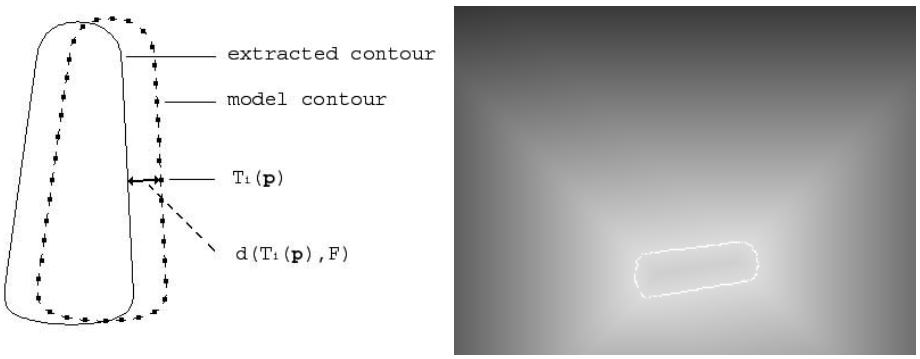
The relative translation and rotation parameters are determined by fitting the model contour to the frame contour. This is achieved by minimizing the cost function for the model's parameters $\boldsymbol{p}$, which measures the approximate distance of model contour to frame contour:

$$E(\boldsymbol{p}) = \sum_{i=1}^{N} d(T_i(\boldsymbol{p}), F) = \sum_{i=1}^{N} D(T_i(\boldsymbol{p})) \tag{4}$$

where $d(T_i(\boldsymbol{p}), F)$ is a distance from a sampled contour point $T_i(\boldsymbol{p})$ to the frame contour F, as in figure 1. The distance $d(T_i, F)$ can be efficiently evaluated by computing a distance transform D on the frame contour image once for every frame, and taking its value $D(T_i(\boldsymbol{p}))$ at the point $T_i(\boldsymbol{p})$. The contour points of a superquadric model are sampled at N equidistant points $T_i(\boldsymbol{p})$, as described in section 2. Levenberg-Marquardt minimization is used for minimizing the cost function and thus evaluating the model's parameters. The gradient of the cost function is computed from model-image Jacobian, as follows:

$$g_E = \sum_{i=1}^{N} \frac{dD(T_i(\boldsymbol{p}))}{d\boldsymbol{p}} = \sum_{i=1}^{N} J_i^T \frac{\partial D}{\partial T_i} \tag{5}$$

The method at present does not include any error recovery possibilities.



(a) (b)

Fig. 1. Computing error of fit: (a) distances from model's contour to the extracted contour can efficiently calculated using distance transform (b)

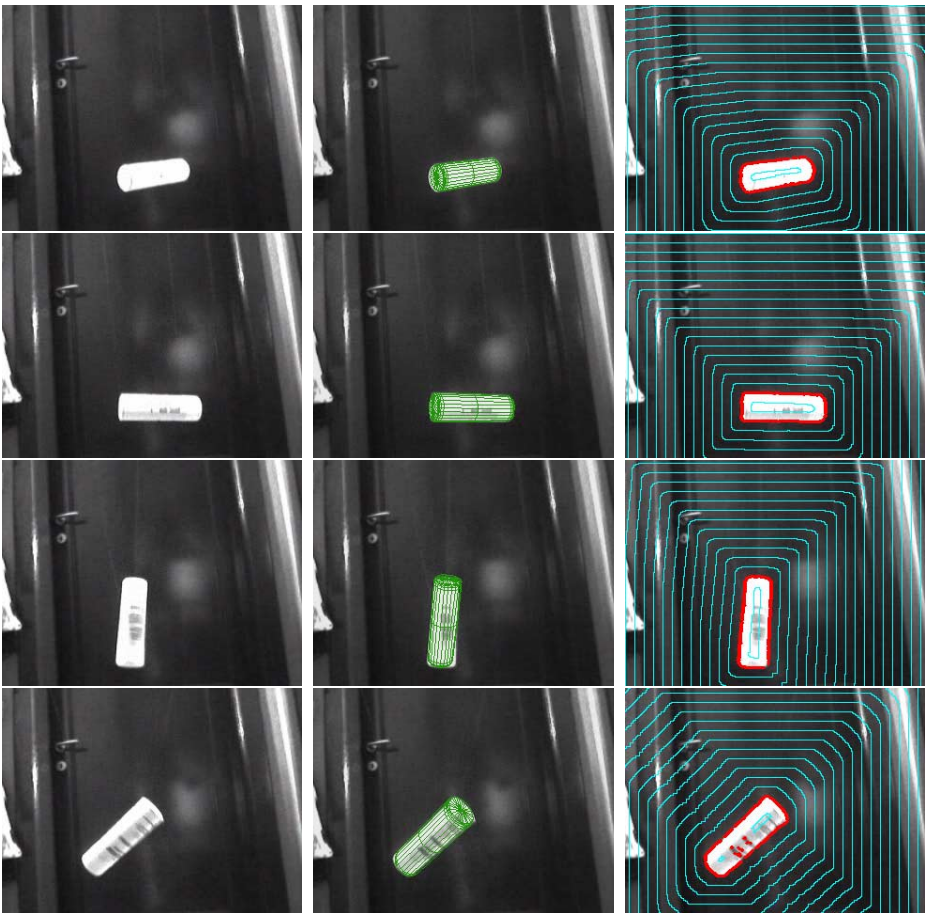
4 Experimental Results

First we tested the method on synthetically produced contours. The contours were produced with the method described in section 2, and the initial superquadric was moved and rotated to different positions away from the contour producing one with an easy-to-use graphical interface. The method showed very good results.

Next the tracking on real video sequences was tried. We present the results of the method in an example tracking. The object to be tracked is a cylindrical shampoo bottle. The bottle was moved in front of the contrast background in order to obtain good contour information. The model was initialized manually and visually. The sequence of 50 320x240 images was taken at 15 frames per second. The computation took 40 to 60 milliseconds per frame on a Pentium Celeron 850 based PC. The results are presented in figure 2.

The results are fairly good, one can observe that the object fits the frame contour well. Due to the contour depth ambiguities, the object is not rotated well in frames 30 and 50. The use of contour information only does not suffice to determine the depth-wise orientation of the object.

The system is dependant on reasonably good contour extraction. The preliminary testing showed, that the most unrecoverable tracking errors are due to bad contour extraction, so more robust contour extraction would lead to much fewer unrecoverable tracking errors. Errors arising from lack of depth information that lead to error in contour interpretation would certainly still be present, even more so when articulated object would be tracked. We believe that a multi-hypothesis approach could take care of most such errors.



(a) (b) (c)

Fig. 2. Tracking a shampoo bottle on a video sequence. Frames 0, 10, 30, 50 presented in rows from top to bottom, respectively. Columns: (a) frame, (b) superimposed superquadric model and (c) frame contour and its distance level sets (lines 10 points apart)

5 Conclusions and Future Work

In this paper tracking of superquadric modelled object over a sequence of images was investigated. The method presented works by fitting the model's projected contour to the object's contour determined in the image, and using model's previous position as the initial guess. The fitting is a minimization of a cost function, which leads to contour overlap. The cost function is minimized for model parameters, i.e. parameters for translation and rotation.

The technique shows promising results on a single superquadric modelled object. Our next obvious goal is to extend it to articulated object tracking, especially human body tracking. With this extension the system should be able to produce high level information about the person, e.g. what/where is he/she doing/going. Also, as mentioned before, the contour extraction method needs further development in order to achieve more robustness. We believe that model-based contour extraction using additional image information (e.g. optical flow) is the way to go.

The method, like any method that tracks from monocular images and does not have a very accurate model of the tracking object, suffers from inaccuracies in the depth estimation. To overcome this problem, the method can easily be extended to multi-view approach. In the obvious approach the cost function could be a compositum of the costs from all views. This approach could be used in applications like interactive rooms, where cameras have a large baseline and vergence angles, and are feeding the system with very different views. On the other hand, in applications like moving robots, with typically two cameras with small camera baseline and vergence angle, the system could extract additional depth information by the use of stereo correlation. This would have to be integrated in the cost function.

Another problem of the method is that at present a recovery from tracking error is not possible. We believe that an integration of a multi-hypothesis approach would take care of the most of tracking errors that arise from depth ambiguities mentioned above as well as other tracking errors (e.g. the ones that are caused by occlusion). The system would have to keep track of a few model hypotheses of the object, giving the best hypothesis as the estimated object at some frame. Poorly evaluated hypotheses would be discarded, while new hypotheses would be gained by first offsetting and then fitting some well evaluated hypotheses. With this kind of approach, even automatic initialization could be achieved: when there is a newly detected motion in the frame, the initial hypotheses would be distributed over the area in motion. In the subsequent frames, these hypotheses would likely evolve into the correct solution.

References

[1] A. Jaklič, A. Leonardis, and F. Solina. *Segmentation and Recovery of Superquadrics*. Kluwer Academic Publishers, Dordrecht, 2000. 1180, 1181

[2] Q. Delamarre and O. Faugeras. 3d articulated models and multi-view tracking with silhouettes. *International Conference on Computer Vision*, pages 716–721, September 1999. Corfu, Greece. 1180

[3] D. M. Gavrila and L. Davis. 3d modelbased tracking of humans in action : A multiview approach. *Conference on Computer Vision and Pattern Recognition*, pages 73–80, June 1996. San Francisco, CA. 1180

[4] N. Jojic, J. Gu, T. S. Huang, and H. Shen. Computer modeling, analysis and synthesis of dressed humans. *IEEE Trans. on Circuits and Systems for Video Technology*, March 1999. 1180

[5] I. Mikic, M. Triverdi, E. Hunter, and P. Cosman. Articulated body posture estimation from multicamera voxel data. *Computer Vision and Pattern Recognition*, 2001. 1180

[6] R. Plankers and P. Fua. Articulated soft objects for video-based body modeling. *IEEE International Conference on Computer Vision*, pages 394–401, 2001. 1180

[7] C. Sminchisescu and A. Telea. Human pose estimation from silhouettes. a consistent approach using distance level sets. *WSCG International Conference for Computer Graphics, Visualization and Computer Vision*, 2002. Chech Republic. 1180, 1181

[8] C. Sminchisescu and B. Triggs. Covariance scaled sampling for monocular 3d body tracking. *IEEE International Conference on Computer Vision and Pattern Recognition*, 1:447–454, 2001. Hawaii. 1180

[9] Y. Zhang and C. Kambhamettu. 3d head tracking under partial occlusion. *Pattern Recognition*, 35:1545–1557, 2002. 1180