Mosaicing the Interior of Tubular Shapes: 3D-model Fitting

Luís Ruivo, José Gaspar, José Santos Victor
Instituto Superior Técnico, Instituto de Sistemas e Robótica, Lisboa, Portugal

***Abstract*— Our framework for mosaicing the interior of a tubular shape involves acquiring images, finding and reconstructing corresponding feature points, fitting a simple 3D model to the reconstructed points, estimating the camera path and dewarping to one single mosaic image. This paper focus on fitting a simple 3D model to the reconstructed points. We propose a 3D model that is based on cylindrical sections, and is useful both for generating simulated data and implementing the fitting procedure.**

I. Introduction

Mosaicing the interior of tubular shapes consists in combining multiple images into a single one (mosaic) representing all the interior. Mosaicing finds applications in e.g. simplifying inspection works on pipelines: mosaics are fast reports of the structures as compared to (eventually long) videos.

In the case of tubular shapes perfectly cylindrical (no curves), watched by cameras perfectly positioned and aligned with the cylinder axis, mosaicing the interior would be just a polar to cartesian dewarping followed by the registration of corresponding image features. Registration would than be just computing an homography [1] between each pair of images.

In our case, we want to consider tubular shapes including straight and curved sections, and to allow free movement of the camera. Hence we start from the traditional idea of reconstructing points of the scene and then focus on fitting a simple 3D model, to the various tube sections, which makes again simple the dewarping step.

In paper is organized as follows: first we do a small review of related work, in section III we propose a geometric model for a tubular shape and detail how to simulate a sequence of pictures taken inside that tube, in section IV we show how to fit the tubular model to real 3D points data, in section V we show some model fitting results and finally we draw some conclusions.

II. Related work

Scene reconstruction is a well known area in computer vision. Reconstructing a scene traditionally starts by selecting in a video-sequence some features that can be corresponded in a robust manner, e.g. the SIFT features [3]. The motion of the camera between consecutive images is than determined for instance by estimating and factorizing the essential matrix [1], assuming that the camera is calibrated.

Currently, scene reconstruction is further improved with SLAM *(Simultaneous localization and Mapping)* or vSLAM *(Visual SLAM)* processes by assuming a smooth camera motion model. In essence, the SLAM and vSLAM algorithms use an autonomous vehicle that starts at an unknown location and then incrementally build a map of the environment while simultaneously uses this map to compute absolute vehicle location (see [2]).

In this paper we assume the scene has be reconstructed, e.g. by one of the previous methods, and we want to fit a simple 3D model to the tubular shape. The 3D model has to be established in a manner that simplifies the dewarping of the images into a mosaic.

III. Tube model and virtual world

We define the tube model as a set of circular sections. Each circular section is defined by its center point, m_i, its radius, r_i, and the normal vector to the circle plane, v_i (see figure 1-top). This model allows representing tubular shapes with straight and curved segments, and variable section-diameters.

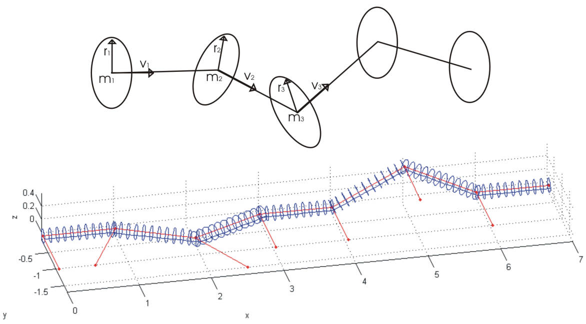

Fig. 1. *(Top)* Tube Model. *(Bottom)* Example of a tubular shape.

The tube model can be used both for creating a virtual world to obtain simulated navigation images and for fitting (representing) real data. Simulating navigation images involves specifying the camera intrinsic parameters and motion. Fitting real data is described in the next section.

IV. 3D model fitting

Given 3D points representing a tubular shape, we fit to that data the simple model defined in the previous section. This fitting process comprises three steps: (i) searching the best fitting cylinders to each of the 3D clouds-of-points (reconstructed between each pair of pair of consecutive images) (ii) removing the overlapping between consecutive cylinders (iii) defining a continuous path along the cylinder sections, i.e.

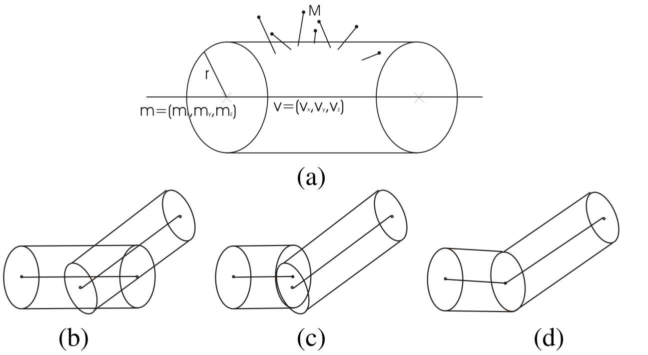

Fig. 2. Fitting cylinders: (a) fit one cylinder to a cloud of points (b) remove the overlapping between cylinders (c) refit cylinders to make the central path continuous (d) final fitting of two cylinders.

matching the ending-face of each cylinder section with the starting-face of the next cylinder section. See figure 2.

Figure 2a represents how a cylindrical shape is fitted to a cloud of points. This fitting corresponds to finding the cylinder yielding the smallest average distance to the observed 3D points. Letting $\{M_i\}$ be the 3D reconstructed points, (m, v, r) the center point of the basis, the axis direction and the radius of the cylinder, then the optimization problem is:

$$\hat{\theta} = \arg_\theta \min \sum_i \left(\| (M_i - m) - proj_v\,(M_i - m) \| - r \right)^2 \tag{1}$$

where θ contains just only a minimal set of degrees of freedom of (m, v, r), namely $\theta = [m_x\ m_y\ v_x\ v_y\ r]$, and $proj_v(x) = x^T vv / \|v\|^2$.

Removing the overlapping between consecutive cylinders is just a truncation of the length of each cylinder such that it ends at the plane defined by basis-face of the next cylinder. The set of non-overlapping cylinders usually does not have a continuous path linking their axis. We enforce this by combining the circles ending and begining consecutive cylinder surfaces into single joining circles. Note that as we allow these circles to have free orientations and radius, we allow for more general shapes (as e.g. sections of cones) instead of just cylinders. See figures 2b, 2c and 2d.

V. Results

A simulated camera was placed traveling inside the tube shown in figure 1-bottom. It travels on the axis of the tube and is always facing the positive Oz direction. Therefore the extrinsic parameters only have a translation component. For this reason, in some images there is no complete view of the tubular section (see figure 3a-right). The camera has 44 degrees field of view and a fixed 640x480 pixel resolution.

The figure 3b represents refitting the tube sections after truncating the cylinders to be non-overlapping. This optimization process adjusts the circles defining the tube sections. By maintaining fixed the first and the fourth circle, the two in the middle are readjusted to keep a continuous/smooth tube. The next iteration maintains the second and the fifth, and the third circle is again readjusted.

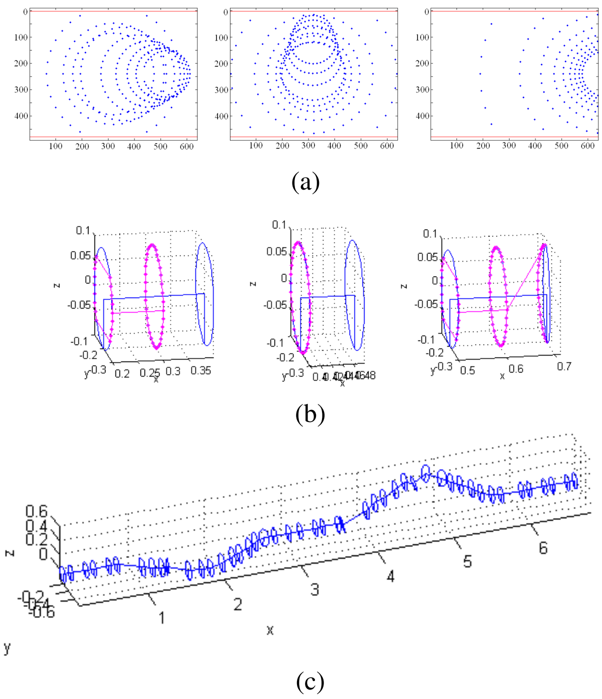

Fig. 3. Tubular shape reconstruction. (a) Three images of the simulated camera inside the tube. (b) Optimizing the connections between sections. *Blue* is the reconstructed tube/circles, *Magenta* is original 3D image points, *Light Blue* is the distance between the original 3D points and the tubular surface. (c) Reconstructed shape.

The figure 3c represents the reconstructed tubular shape. The structure is close to the original despite the local removal of sections overlapping and refitting procedures.

VI. Conclusion

The proposed 3D model based on circles defining tubular sections, is a convenient representation as it allows optimizing locally the fitting process. It is also a useful representation for dewarping as it allows defining a *ground line* as the intersection of the circles with a vertical plane. The line will than be the center line of the mosaic image and the circles are broken at the top and rollover to a 2D plane that will be the 2D image mosaic.

As future work, we plan to implement all the steps of the mosaicing, from the image processing, passing by reconstruction, model fitting (as proposed here), till dewarping the tubular shape and mapping the image data in the mosaic.

Acknowledgement

This work has been partially supported by the POS_C financing program, that includes FEDER funds, by ISR-Lisbon theme-B, and by the European project URUS FP6-EU-IST-045062.

References

[1] Richard Hartley and Andrew Zisserman. Multiple view geometry in computer vision. *Cambridge University Press*, 2000.

[2] N. Karlsson, E. Di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. Munich. The vslam algorithm for robust localization and mapping. *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 24 – 29, 2005.

[3] David Lowe. Distinctive images features from scale-invariant keypoints. *International Journal of computer Vision*, 2004.