

Projective Reconstruction of Surfaces of Revolution

Sven Utcke¹ and Andrew Zisserman²

¹ Arbeitsbereich Kognitive Systeme, Fachbereich Informatik,
Universität Hamburg, Germany
`utcke@informatik.uni-hamburg.de`

² Visual Geometry Group, Department of Engineering Science,
University of Oxford, United Kingdom
`az@robots.ox.ac.uk`

Abstract. This paper addresses the problem of recovering the generating curve of a surface of revolution from a single uncalibrated perspective view, based solely on the object's outline and two (partly) visible cross-sections. Without calibration of the camera's internal parameters such recovery is only possible up to a particular transformation of the true shape. This is however sufficient for 3D reconstruction up to a 2DOF transformation, for recognition of objects, and for transfer between views. We will describe the basic algorithm and show some examples.

1 Introduction

This paper describes a method for the recovery of the generating function of a Surface of Revolution (SOR) from a single image. Ignoring the usual unknowns scale and position the generating function could be used for a reconstruction up to a particular subgroup of 3D-projective transformations with only 2 degrees of freedom (DOF). Additionally, the generating function can be used directly for recognition of the SOR from an arbitrary view point, provided the method of matching plane curves is invariant to a projective transformation. Transfer of the contour between views finally can be used for verification.

Most algorithms for the reconstruction of SORs (or, more generally, Straight Homogeneous Generalised Cylinders, SHGCs) go back to [1], where a possible algorithm for the identification of a SHGC's axis from its outline was given, on to an algorithm for the identification of a SHGC's ending cross-sections in [2], and [3,4,5,6] who all gave algorithms for the reconstruction of SHGCs from orthographic views (e. g. approximated by a tele-lens). For SORs the approach in [5,6] was recently extended to work with any calibrated camera [7]. All of these algorithms require knowledge about the actual camera used, and mostly even a particular camera geometry, in addition to the object's outline and at least one (partly) visible cross-section. The cross-section could be a discontinuity in the SOR's generating function such as the flat top or bottom of an object, or surface markings. Our novel algorithm by contrast requires the outline and two partly visible cross-sections, but no additional information about the imaging process

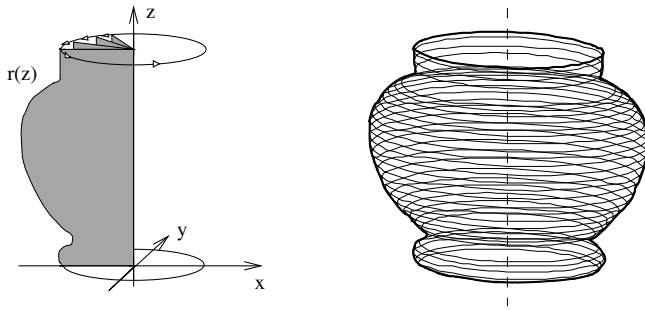


Fig. 1. SORs as either a rotational surface or as a stack of circular cross-sections.

— it is in fact invariant not only to perspective transformation, but the entire projective group.

Other methods for the reconstruction of SORs previously employed by us were e.g. based on a number of distinguished points [8,9]. In [10] we gave an algorithm for the computation of an SOR's projectively (quasi-) invariant signature and in [11] these methods demonstrated a recognition-rate similar to that of appearance-based approaches. Since we already demonstrated in [12,6] that the identification and grouping of potential SORs in cluttered images is possible, we will in this paper concentrate on simple images so as not to confound the issues.

This paper is structured as follows: we will recall the basic properties of an SOR in Sec. 2; Section 3 gives a description of the actual algorithm used for the projective reconstruction as well as some examples, and applications to recognition and transfer will be discussed in Sec. 4.

2 Object Model

There are two traditional models for the construction of Surfaces of Revolution. Most commonly used is that of a *generating function* $r(z)$ being rotated around the axis of revolution, resulting in a surface

$$\mathbf{S} = (r(z) \cos(\phi), r(z) \sin(\phi), z)^T, \quad (1)$$

compare Fig. 1. Our intentions, however, are best served if we understand an SOR as a special case of a Straight Homogeneous Generalised Cylinder, namely one with a circular cross-section where the axis goes through the centre of the cross-section, and the cross-section is orthogonal to the axis. In such a model the circular cross-section is swept along the SOR's axis and scaled according to a *scaling function* (equivalent to the planar generating function mentioned above); further simplifying this model a little, we can imagine an SOR to be constructed out of (infinitely) many circular cross-sections stacked on top of each other. Figure 1 illustrates both models. The idea of forming the SOR outline as an envelope of circles is the backbone of the novel method presented here.

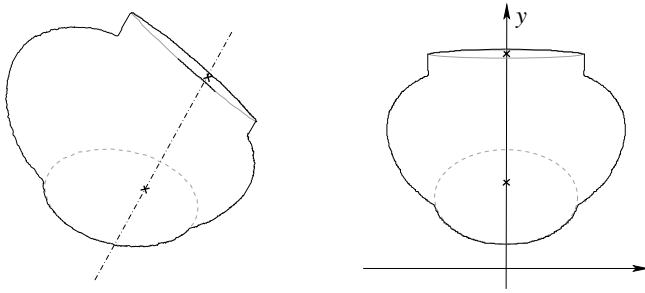


Fig. 2. Projectively distorted outline of an SOR in arbitrary position, and the rectified version symmetric around the y -axis

3 Projective Reconstruction

In this section we will show how, from a single image, an SOR's generating function $r(z)$ can be recovered up to a 2DOF subgroup of the plane projective transformations (discounting the usual overall scale ambiguity and an offset along the axis). All that is needed is the image of an SOR's outline and two of its cross-sections, C_1 and C_2 , two conics.

The algorithm proceeds by first identifying and grouping possible SORs in the image and computing a transformation into a rectified frame where the outline as well as the image of the two cross-sections C_1 and C_2 will be symmetric around the y -axis; this can always be done using the algorithms described in [10,12, 6] and will not be discussed in any detail. From there a planar homography into a distinguished frame is found where the image of each cross-section will be a circle; this immediately results in a projective reconstruction $\varrho(y)$ of the generating function $r(z)$. We will now describe the process in more detail, but touching only lightly on most of the previously published results mentioned above.

The identification and grouping of an SOR's outline draws heavily on the fact that the two sides of its outline (i. e. on both sides of the projection of the axis) are related by a planar harmonic homology, which can be calculated quite accurately using one of [10,12]. Once this homology has been found it is always easily possible to rectify the outline in such a way that it will be symmetric around the y -axis of a 2D-plane. Figure 2 gives an example of a severely distorted image of an SOR and its rectified version, which is symmetric around the y -axis.

The outline or contour of an SOR is the image of the so called *contour generator*, the set of points where rays through the viewpoint are tangent to the object; this will generally be a space-curve in 3D, and no straightforward invariant transformation exists between the outline as observed in the image and the generating function. For each individual cross-section however such a transformation is easily found, and since all cross-sections were originally completely contained within the SOR they will, after projection, be completely contained

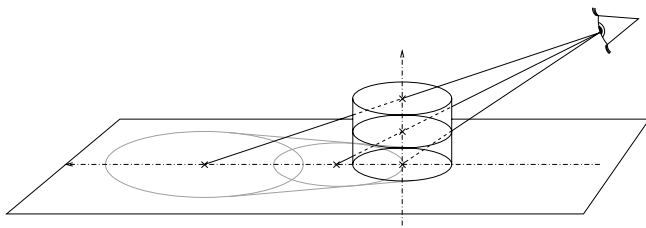


Fig. 3. Independent of viewpoint a planar homography from the image plane onto a distinguished plane exists such that the images of all cross-sections become circles

within the SOR’s projected 2D-outline — this means that the outline is nothing else but the envelope of all these projected cross-sections.

For each view the position of the centre of projection uniquely determines the view-cone and therefore the contour-generator. However, it does not determine the position of the image-plane, and it is possible to generate the information on any such plane from the information on any other plane by a simple plane to plane projective transformation, a so called planar homography. This means that we are free to put a virtual image plane wherever we like. There exist in particular a set of distinguished image planes that maintain the y symmetry and where the projection of each circular cross-section will again be a circle; these are the planes which are parallel to the cross-sections in 3D, and in particular any plane containing a cross-section. In Fig. 3 this is illustrated for a cylinder.

How do we find the homography from the image-plane into one of these distinguished planes? Since the rectified outline is symmetric around the y -axis this must also hold for the images of the two cross-sections, which can therefore be parameterised as conics of the form $\mathbf{x}^T \mathbf{C} \mathbf{x} = 0$ with

$$\mathbf{C}_i = \begin{pmatrix} 1 & 0 & 0 \\ 0 & C_i & \frac{E_i}{2} \\ 0 & \frac{E_i}{2} & F_i \end{pmatrix}. \tag{2}$$

The homography we are looking for needs to apply some projective skew in y -direction such that the aspect ratios of both conics will be the same, as well as some anisotropic scaling along one of the two axes (we choose the x -axis) such that both conics will become circles, it therefore has the form

$$\mathbf{H} = \begin{pmatrix} s & 0 & 0 \\ 0 & 1 & 0 \\ 0 & b & 1 \end{pmatrix}. \tag{3}$$

If $\mathbf{x}' = \mathbf{H} \mathbf{x}$ is the image of the point \mathbf{x} under the homography \mathbf{H} , then

$$\mathbf{C}'_i = \mathbf{H}^{-T} \mathbf{C}_i \mathbf{H}^{-1} = \begin{pmatrix} \frac{1}{s^2} & 0 & 0 \\ 0 & C_i - bE_i + b^2F_i & \frac{E_i}{2} - bF_i \\ 0 & \frac{E_i}{2} - bF_i & F_i \end{pmatrix} \tag{4}$$

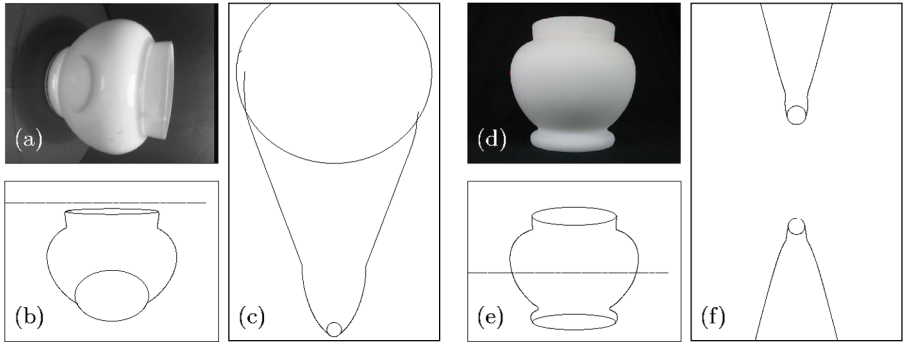


Fig. 4. Original photographic image (a), (d); outline, conics, and the conics’ vanishing line (b), (e); outline and conics transferred onto a distinguished plane (c), (f). Note that (a) displays serious non-linear distortions, which have a negative impact on the calculations — the top conic appears to sit above the object in (c)

is how the two conic-cross-sections C_1 and C_2 will transform, and based on our requirement that both conics should have the same aspect-ratio after transformation, we can calculate two possible values for b as

$$b_{I/II} = \frac{(E_1 - E_2) \pm \sqrt{(E_1 - E_2)^2 - 4(C_1 - C_2)(F_1 - F_2)}}{2(F_1 - F_2)}. \tag{5}$$

The line $\ell = (0, 1, 1/b)^T$ is the planes’ vanishing line within the rectified image. The two different values for b code the two different cases where either both conics are viewed from the same side, i. e. the viewpoint is completely above or below both conics, and the line ℓ will be outside the object, and the case where one conic is viewed from above and one from below, i. e. the viewpoint is somewhere between the two conics, and so is the line ℓ , which intersects the object. Figure 4(a)+(d) show two pictures of real objects, each corresponding to one of the two cases for b in (5), as well as the objects’ outlines, conics and the lines $\ell = (0, 1, 1/b)^T$ in Fig. 4(b)+(e).

Once b has been found it is then easy to find a solution for s based on the stipulation that the conics should be circles after scaling; this solution is

$$s = (C_i - bE_i + b^2F_i)^{-1/2}. \tag{6}$$

Since the knowledge of b and s uniquely determines the homography H in (3) we can next transfer the outline into this distinguished plane. We also know that here the images of all cross-sections are circles and that the outline is the envelope of all these circles; we can therefore in reverse find the images of all intermediate cross-sections as circles which are 1) fully contained within the outline, have 2) their centres on the image of the SOR’s axis, and are 3) tangent at two corresponding points of the outline. The left image of each set in Fig. 5

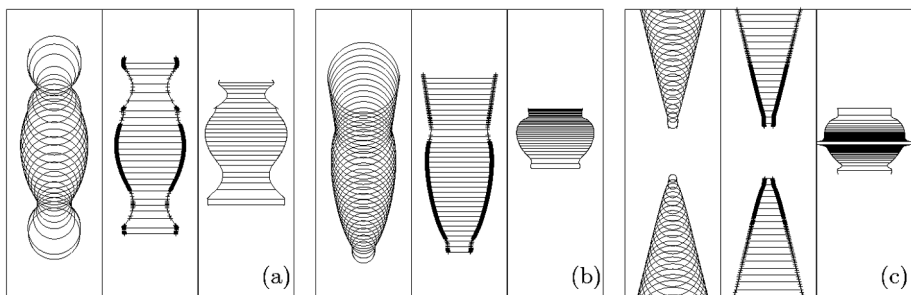


Fig. 5. The outline of an SOR as the envelope of cross-sections (*left* image of each set), a projective reconstruction of the generating function (*middle* image of each set), and an alternative representation of the generating function (*right*)

shows the image of an SOR’s outline on the distinguished plane, as well as some of the inscribed cross-sections.

Once the centre $(0, y)^T$ and radius ϱ have been computed for each circle we immediately have a projective reconstruction $\varrho(y)$ of the generating function. From Fig. 3 it can be seen that y is related to the true distance along the axis z by a line to line projective transformation; the complete 3×3 plane projective transformation can be represented as $(r, z, 1)^T = \mathbf{G} \cdot (\varrho, y, 1)^T$ with

$$\mathbf{G} = \begin{pmatrix} a & 0 & 0 \\ 0 & as_y & t_y \\ 0 & c & 1 \end{pmatrix} \tag{7}$$

which is of the same basic form as (3) and — ignoring the usual overall scale ambiguity and choice of origin ($a = 1$ and $t_y = 0$) — has 2 DOF, which code the vertical vanishing point (in c) and aspect-ratio (in s_y). Examples are shown in the middle image of each set in Fig. 5.

However, while being projectively equivalent to the original generating function it will in general not be very similar by human standards — this is particularly true for Fig. 5(c). We have therefore used the inverse of (3) to project the reconstructed outline back into the rectified image, where the projective reconstruction will generally look more realistic. The right image of each set in Fig. 5 shows this projection. How close these are to an Euclidean reconstruction depends on the viewing-position. For near orthographic projections ($c = 0$) the reconstruction will already be true up to anisotropic scaling, but in that case we could have used one of [5,6], requiring only one cross-section instead of two.

4 Applications

The recovered generating function can be used to reconstruct the surface up to the particular projective ambiguity \mathbf{G} given in (7) above. However, it can also be

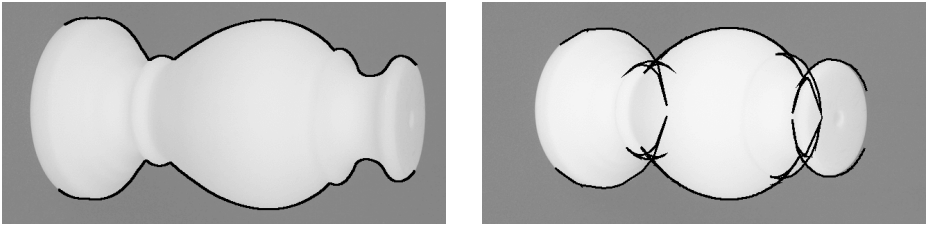


Fig. 6. Examples of transfer from one particular view of an SOR into a second view

used directly (despite the projective ambiguity) for two other tasks: recognition and transfer, and we describe these now.

Ignoring self-occlusion, the recovered generating curve is within a projective transformation of the true generating curve, so any method for projectively invariant curve representation may be used to recognise it. This is more powerful than the distinguished points previously used by us [8,9], since the entire generating curve can be matched rather than just the isolated points available on the original outline; a number of possible representations are cited in [13,14,15].

The result could then e.g. be verified using *transfer*, i.e. given an SOR's outline and two cross-sections in one image, and the corresponding two cross-sections in a second image, we are trying to predict the contour in that second image; the predicted and measured outline have to agree. Although for each view the generating function can only be recovered up to a 2 (4) DOF ambiguity as described in (7), it is still possible to compute the map between the generating function and the image contour for any view in which the two cross-sections have been specified, using the construction of Sec. 3 in reverse, generating the contour as the envelope of cross-sections. The position and radii of the two cross-sections serve to uniquely compute the missing parameters in (7).

Figure 6 gives an example of the transfer of a contour from one view of an SOR into a different view of the same SOR, this demonstrates nicely the correctness of the reconstruction. The cusps on the second contour (as, indeed, all contours inside the object-outline) correspond to parts of the contour generator which, for opaque objects, can not be observed in that second, slightly more tilted view. This effect is called self-occlusion. The part of the contour generator corresponding to the self-occluded bit of the contour can never be reconstructed from contour-information alone (although it might be possible to use grey-level cues) and will consequently be missing from the reconstruction, something which an algorithm for the recognition of SORs would need to take into account.

5 Conclusion

We have demonstrated a simple method which, based on the outline of an SOR and two cross-sections, can easily create a projective reconstruction for all parts of the generating function where the object is not (self-) occluded. This method

does not require any knowledge about the camera but does in fact work for an arbitrary projective transformation of the contour. The resulting reconstruction can then be used to recognise the object by any of a number of projectively invariant approaches, or to calculate the transfer into additional views.

References

1. Ponce, J., Chelberg, D., Mann, W.B.: Invariant properties of straight homogeneous generalized cylinders and their contours. *IEEE Trans Pattern Anal Mach Intell* **11** (1989) 951–966
2. Sato, H., Binford, T.O.: On finding the ends of SHGCs in an edge image. In: *Image Understanding Workshop, San Diego, CA, DARPA, Morgan Kaufmann Publishers, San Mateo, CA* (1992) 379–388
3. Gross, A.D., Boulton, T.E.: Recovery of SHGCs from a single intensity view. *IEEE Trans Pattern Anal Mach Intell* **18** (1996) 161–180 errata in [16].
4. Sato, H., Binford, T.O.: Finding and recovering SHGC objects in an edge image. *Comput Vision, Graphics & Image Processing: Image Understand* **57** (1993) 346–358
5. Zerroug, M., Nevatia, R.: Volumetric descriptions from a single intensity image. *Int J Comput Vision* **20** (1996) 11–42
6. Abdallah, S.M., Zisserman, A.: Grouping and recognition of straight homogeneous generalized cylinders. In: *Asian Conf Comput Vision, Taipei* (2000) 850–857
7. Wong, K.Y.K., Mendonça, P.R.S., Cipolla, R.: Reconstruction of surfaces of revolution from single uncalibrated views. In: *Brit Mach Vision Conf.* (2002) 93–101
8. Forsyth, D.A., Mundy, J.L., Zisserman, A., Rothwell, C.A.: Recognising rotationally symmetric surfaces from their outlines. In Sandini, G., ed.: *Proc Eur Conf Comput Vision. LNCS, Springer Verlag* (1992) 639–647
9. Liu, J., Mundy, J., Forsyth, D., Zisserman, A., Rothwell, C.: Efficient recognition of rotationally symmetric surfaces and straight homogeneous generalized cylinders. In: *Proc Conf Comput Vision Pattern Recognit, New York City, New York, USA, IEEE CS, IEEE CS Press* (1993) 123–128
10. Pillow, N., Utcke, S., Zisserman, A.: Viewpoint-invariant representation of generalized cylinders using the symmetry set. *Image Vision Comput* **13** (1995) 355–365
11. Mundy, J., Liu, A., Pillow, N., Zisserman, A., Abdallah, S., Utcke, S., Nayar, S., Rothwell, C.: An experimental comparison of appearance and geometric model based recognition. In: *Proc. Object Representation in Computer Vision II. LNCS 1144, Springer-Verlag* (1996) 247–269
12. Zisserman, A., Mundy, J., Forsyth, D., Liu, J., Pillow, N., Rothwell, C., Utcke, S.: Class-based grouping in perspective images. In: *Proc Int Conf Comput Vision, Cambridge, MA, USA, IEEE CS, IEEE CS Press* (1995) 183–188
13. Carlsson, S., Mohr, R., Morin, L., Rothwell, C., Van-Gool, L., Veillon, F., Zisserman, A.: Semi-local projective invariants for the recognition of smooth plane curves. *Int J Comput Vision* **19** (1996) 211–236
14. Reiss, T.H.: Object recognition using algebraic and differential invariants. *Signal Processing* **32** (1993) 367–395
15. Holt, R.J., Netravali, A.N.: Using line correspondences in invariant signatures for curve recognition. *Image Vision Comput* **11** (1993) 440–446
16. Gross, A.D., Boulton, T.E.: Correction to “recovery of SHGCs from a single intensity view”. *IEEE Trans Pattern Anal Mach Intell* **18** (1996) 471–479