Article:
Puech, W, Bors, A G, Pitas, I et al. (2001) Projection distortion analysis for flattened image mosaicing from straight uniform generalized cylinders. Pattern recognition. pp. 1657-1670. ISSN: 0031-3203

https://doi.org/10.1016/S0031-3203(00)00056-X

Projection Distortion Analysis for Flattened Image Mosaicing from Straight Uniform Generalized Cylinders

William PUECH[1], Adrian G. BORŞ[2],

Ioannis PITAS[3], and Jean-Marc CHASSERY[4]

1 Modelisation and Signal Laboratory, University of Toulon,

Av. G. Pompidou, BP 56, 83162 La VALETTE DU VAR Cedex, FRANCE - puech@univ-tln.fr

2 Department of Computer Science, University of York, York YO10 5DD, U.K. - Adrian.Bors@cs.york.ac.uk

3 Department of Informatics, University of Thessaloniki, Box 451,

54006 THESSALONIKI, GREECE - pitas@zeus.csd.auth.gr

4 TIMC-IMAG Laboratory, Institut Albert Bonniot, Domaine de la Merci,

38706 LA TRONCHE Cedex, FRANCE - Jean-Marc.Chassery@imag.fr

Abstract

This paper presents a new approach for reconstructing images mapped or painted on straight uniform generalized cylinders (SUGC). A set of monocular images is taken from different viewpoints in order to be mosaiced and to represent the entire scene in detail. The expressions of the SUGC's projection axis are derived from two cross-sections projected onto the image plane. Based on these axes we derive the SUGC localization in the camera coordinate system. We explain how we can find a virtual image representation when the intersection of the two axes is matched to the image center. We analyze the perspective distortions when flattening a scene which is mapped on a SUGC. We evaluate the lower and the upper bounds of the necessary number of views in order to represent the entire scene from a SUGC, by considering the distortions produced by perspective projection. A region matching based mosaicing method is proposed to be applied on the flattened images in order to obtain the complete scene. The mosaiced scene is visualized on a new synthetic surface by a mapping procedure. The proposed algorithm is used for the representation of mural paintings located on SUGC's with closed cross-sections (circles for columns), or opened cross-sections (ellipses or parabolas for vaults).

Keywords

Monocular vision, straight uniform generalized cylinder, cross-sections, localization, perspective projection, image mosaicing, visualization.

1 Introduction

When taking a picture of a scene mapped on a straight uniform generalized cylinder (SUGC), the scene is distorted due to the perspective projection on the image plane. If the curved surface is convex, it is not possible to obtain the entire surface representation in only one image. In this study, we consider various images taken from different viewpoints located around a SUGC, in order to obtain the reconstruction of the complete scene by projecting, flattening and mosaicing it.

In monocular vision, in order to backproject an image on the SUGC we need to find the localization parameters represented by three rotation angles, each corresponding to an axis of the 3-D coordinate space. Localization has been performed by interpreting a triplet of image lines as the perspective transformation of a triplet of linear ridges of an object model [1] or by using the zero-curvature points of contours from an image [2]. It is also possible to determine the location and the orientation of a cylinder with a label on it [3] or by using the light source direction [4]. When the surface model is known, it is easy to project it on the image plane. By using the correspondence of this projection with the original image we obtain the localization parameters [5]. A flattening method based on genetic algorithms was proposed in [6].

For each image taken from a different viewpoint, we derive the localization based on a certain *a priori* knowledge. After detecting the projections of two cross-sections from the SUGC on the image, we calculate their common normal and afterwards we find their localization parameters. Based on the localization parameters, we backproject the image on the SUGC and, afterwards, we flatten the surface of SUGC. The result is an image containing a part of the scene without geometrical distortions caused by the surface curvature. However, certain geometrical distortions are caused by the perspective view. Segments of different sizes from the cylinder surface project in image segments of identical size (pixels). The variation in perspective distortion was used for computing the shape of curved surfaces from texture information in [7]. In this paper we provide an analysis of the perspective projection distortions in the case of a scene mapped or painted on a SUGC with circular cross-sections (SUGCC). We derive a formula which quantifies the distortion caused in the image with respect to the distance from the axis of the SUGCC and the cross-section radius.

Mosaicing is a technique which assembles a set of images for obtaining a general accurate representation of the entire scene [8]. After flattening, we employ mosaicing in order to reconstruct the entire image of the SUGC surface. A classical approach adopted in image mosaicing is based on selecting a set of corresponding points on each pair of images [9, 10]. Image mosaicing was employed for reconstructing scenes from video sequences [11]. The overlapping among the images to be mosaiced in these applications

is considered to be large and the image mosaicing problem is treated similarly to the image registration approach [12]. In [12] a wavelet approach is considered for image registration while in [11], the Levenberg-Marquardt algorithm is used for the minimization of an error function between successive images in video mosaics. In [13] the perspective projection distortions in the images taken with a camera which rotates around its focal point are corrected by projecting the images onto a Gaussian sphere. In this study, we propose a new mosaicing algorithm based on matching the common regions from two given images representing neighboring parts of the surface. The matching approach is similar to the block matching algorithms, extensively used for estimating the optical flow in image sequences [14]. However, the regions corresponding to the SUGC surface margin projection after flattening, may contain severe distortions due to perspective projection [15]. The regions which contain big distortions are not appropriate to be considered in the matching algorithm and they are not taken into account by the matching algorithm. Pixels from overlapping areas are calculated as a weighted sum of the corresponding pixel values in the original images. The proposed algorithm is applied in painting representation. After mosaicing, the resulting image can be used for painting restoration [16].

In Section 2 the localization of the SUGC is described. The theoretical analysis of the geometrical distortions and the evaluation of the necessary number of views are provided in Section 3. In Section 4 we show how we obtain a scene without distortions after flattening and mosaicing. We apply the proposed technique in mural painting and ceramic art visualization in Section 5.

2 The straight uniform generalized cylinder localization

In this Section we describe how we can find the SUGC localization from a single perspective view. The localization is derived based on the three rotation angles θ_x, θ_y and θ_z. These rotation angles give us the relationship between the camera coordinate system and the SUGC coordinate system [17]. The localization is done separately for each view. Based on the localization parameters we backproject the image on the 3-D surface.

Two projection axes must be localized [18, 19]. In Section 2.1 we show how to find the projection of the SUGC axis. and in Section 2.2 how to find the position of the second axis corresponding to the projection of one particular cross-section. Based on the equations of the two axes we obtain three rotation angles θ_x, θ_y and θ_z describing the position of the object $(O', \vec{i}, \vec{j}, \vec{k})$, in the camera coordinate system $(O, \vec{x}, \vec{y}, \vec{z})$, where O is the viewpoint, as shown in Figure 1. In Section 2.3 we show how to obtain a new image

(called virtual image) by rotating the camera with the given angles.

2.1 Finding the projection of the SUGC axis

In order to find the SUGC axis in the image we use a certain *a priori* knowledge. We detect two curves in the projected image which are projections of cross-sections located on the SUGC. Examples of a painted SUGC surface are shown in Figure 2. Many studies have already been done for finding the projection of the SUGC axis, e.g., by using mathematical morphology [20], by finding local symmetries [21] or by using a method based on expectation-maximization [22].

In our approach, after the detection of the two curves we find the projection of the SUGC axis in the image [23]. In order to do this, we identify the common normal P_1P_2 of the two cross-section projections in the image as shown in Figure 3. We use an iterative method consisting of successive derivations of normals to the curves [19]. This method stops when two successively estimated normals on the same curve are very close to each other. Thus, we identify two points P_1 and P_2. The slope of the straight line $P_1\,P_2$ gives us the direction of the axis [18]. In the case of a convex surface, the projection of this axis in the image corresponds to the nearest meridian to the viewpoint. In the coordinate system of the image (u, v), the equation of this axis is :

$$v = A_1.u + B_1, \tag{1}$$

where A_1 and B_1 are the coefficients of the straight line determining this axis. From equation (1) we derive immediately the rotation angle θ_z. This rotation angle corresponds to the angle between u and the SUGC axis, as shown in Figure 3 :

$$\theta_z = \arctan(A_1). \tag{2}$$

2.2 The derivation of the second axis

Among all the cross-sections of the SUGC, only one can be projected on the image as a straight line. This straight line belongs to the plane passing through the cross-section and the viewpoint. Its corresponding set of surface points is defined as the second axis [18] while the first axis corresponds to the SUGC axis as described in Section 2.1. In the following, we explain how to find the second axis in the image in order to obtain the rotation angles, θ_x and θ_y. The second axis is orthogonal to the axis defined by (1) as shown in Figure 3. Let us consider P, a point located on the curve passing through

either P_1 or P_2. We define the curvature at a point P_i as :

$$K_i = \lim_{P \to P_i} \frac{\alpha(P) - \alpha(P_i)}{|\widehat{PP_i}|}, \quad (3)$$

where $i \in \{1, 2\}$, $\alpha(P_i)$ is the angle of the tangent to P_i, and $|\widehat{PP_i}|$ is the length of the arc between P and P_i. If K_1 and K_2 have different signs, we can assume that the curvature changes linearly throughout the axis P_1P_2. For each curve, the normal line at the intersection point with the projection of the SUGC axis coincides with the common normal (1).

We define by P_0 the point belonging to the SUGC axis where the curvature K_0 equals to zero. By including such information in equation (1), we obtain the equation of the second axis :

$$v = -\frac{1}{A_1}.u + \left(v_0 + \frac{u_0}{A_1}\right), \quad (4)$$

where (u_0, v_0) are the coordinates of the image center. We denote by (Δ_u, Δ_v) the vector distance between (u_0, v_0) and P_0, the intersection of the two axes. The two rotation angles are then given by :

$$\begin{cases} \theta_x = \arctan(\frac{\Delta_v}{f.s}) \\ \theta_y = \arctan(\frac{\Delta_u}{f.s}), \end{cases} \quad (5)$$

where f is the focal distance and s is the resolution factor.

2.3 The construction of the virtual image

From the results obtained in the previous Section we are able to localize the SUGC in the camera coordinate system [23]. Based on the camera rotation angles $(\theta_x, \theta_y, \theta_z)$ given by the equations (2) and (5), we can match the point P_0 with the image center (u_0, v_0). Then, we obtain the virtual image as shown in Figure 4. The original image is projected onto the virtual image. In the virtual image, P_0 is projected to the image center and the projection of the SUGC axis is vertical.

Let us denote by $z = f$, the equation of the image plane in the original camera coordinate system. The coordinates of the normal vector $\vec{v}$ to this plane are $(0, 0, 1)$. Let us denote by M, an arbitrary point from the original image. We have $\vec{OM} \cdot \vec{v} = f$. Thus, the image plane is defined by the focal distance f and by the normal vector $\vec{v}$. The virtual image is defined by the focal distance f' and by the normal vector $\vec{v'}$:

$$\vec{OM'} \cdot \vec{v'} = f', \quad (6)$$

where M' is the point corresponding to M in the virtual image. Let us consider the virtual image plane obtained by rotating the original image :

$$\begin{cases} \vec{OM'} = \mathbf{R}\,\vec{OM} \\ \vec{v'} = \mathbf{R}\vec{v}\,, \end{cases} \tag{7}$$

where f' is the focal distance equal to f, and $\mathbf{R}$ is a 3×3 rotation matrix according to the rotation angles $(\theta_x, \theta_y, \theta_z)$. Finally, we obtain :

$$\begin{cases} f' = f & = & \vec{OM} \cdot \vec{v} \\ \vec{v'} & = & \mathbf{R}\,\vec{v} \end{cases} \tag{8}$$

From the rotation matrix $\mathbf{R}$ we derive $\vec{v'}$:

$$\vec{v'} = \begin{pmatrix} \cos\theta_z \sin\theta_y \cos\theta_x - \sin\theta_z \sin\theta_y \\ \sin\theta_z \sin\theta_y \cos\theta_x - \cos\theta_z \sin\theta_x \\ \cos\theta_y \cos\theta_x \end{pmatrix}. \tag{9}$$

The equation of the original plane in this new camera coordinate system is :

$$(\cos\theta_z \sin\theta_y \cos\theta_x - \sin\theta_z \sin\theta_y)x' + (\sin\theta_z \sin\theta_y \cos\theta_x - \cos\theta_z \sin\theta_x)y' + (\cos\theta_y \cos\theta_x)z' = f. \tag{10}$$

The equation of the virtual image plane is $z' = f$ and the image is obtained by perspective projection from the original image.

3 Representation of the scene located on a SUGC with circular cross-sections

In the previous Section we have shown how we can obtain a virtual image for an arbitrary camera coordinate system. Certain distortions which are caused due to perspective projection can be observed in SUGC images. Let us consider the virtual image of a SUGC with circular cross-sections (SUGCC). In this Section we quantify the level of distortion in the image representing a SUGCC. Afterwards, based on this analysis we estimate the number of images that should be taken around the SUGCC in order to represent its entire surface.

3.1 Analysis of geometrical distortions caused by perspective projection

A representation of a horizontal cross-section through a SUGCC and through the virtual image plane is shown in Figure 5. In this representation, the camera coordinate system has an axis parallel with the SUGCC's axis. The focal axis z is perpendicular to the SUGC's revolution axis. The radius of the cross-section is r and the viewpoint O is located at a distance l from the SUGCC axis. The projection of the arc $|\widehat{A_1 A_{2m}}|$ to the virtual image plane is the line segment $|U_1 U_{2m}|$.

The horizontal cross-section through the image is shared in $2m$ equally sized intervals corresponding to pixels :

$$|U_k U_{k-1}| = |U_{k-1} U_{k-2}| = \ldots = |U_2 U_1| = 1\,\text{pixel}. \tag{11}$$

Each arc of the cross-section, denoted as $|\widehat{A_k A_{k-1}}|$ is projected on the image in a segment $|U_k U_{k-1}|$, $k = 2, \ldots, 2m$. Let us denote by α_k, the view angle $\widehat{A_1 O A_k}$ corresponding to an image segment $|U_1 U_k|$. The angle from the SUGCC's axis corresponding to the arc $|\widehat{A_1 A_k}|$ is denoted by $\widehat{A_1 O' A_k} = \beta_k$.

In the following, we evaluate the angle β_k as a function of given geometrical parameters. Let us consider $|A_k D|$ the perpendicular to $|A_1 O|$ from the point A_k of the SUGCC's horizontal cross-section. We can express the length of the line segment $|A_k D|$ in two different ways. From classical geometry properties we have :

$$|A_1 A_k| = 2r \sin\left(\frac{\beta_k}{2}\right). \tag{12}$$

From the triangles $O' A_1 A_k$ and $A_1 A_k D$ we derive :

$$\frac{|A_k D|}{r \sin\left(\frac{\beta_k}{2}\right)} = \frac{2r \sin\left(\frac{\beta_k}{2}\right)}{r} \tag{13}$$

and we obtain

$$|A_k D| = 2r \sin^2\left(\frac{\beta_k}{2}\right). \tag{14}$$

We observe that in triangle $A_1 O' O$:

$$|A_1 O| = \sqrt{l^2 - r^2} \tag{15}$$

$$|A_1 D| = r \sin(\beta_k) \tag{16}$$

$|A_k D|$ is calculated from the triangle $A_k D O$ as :

$$|A_k D| = (\sqrt{l^2 - r^2} - r \sin \beta_k) \tan \alpha_k. \tag{17}$$

Let us denote by μ the ratio between the distance l from the viewpoint to the SUGCC axis, and the SUGCC cross-section radius r :

$$\mu = \frac{l}{r}. \tag{18}$$

From the triangle OU_kU_m we have :

$$\tan(\alpha_m - \alpha_k) = \frac{|U_mU_k|}{|U_mO|} \tag{19}$$

Based on classical geometry properties in the triangles OU_1U_m, $OO'A_1$ and by using (11) we find :

$$\tan(\alpha_m - \alpha_k) = \frac{|U_mU_k|}{|U_1U_m|}\frac{|U_1U_m|}{|U_mO|} = \frac{m-k}{m}\frac{|O'A_1|}{|A_1O|} = \frac{m-k}{m\sqrt{\mu^2-1}} \tag{20}$$

where α_m represents the angle $O'\widehat{O}A_1$ in Figure 5. From the triangle $OO'A_1$ we have :

$$\tan\alpha_m = \frac{1}{\sqrt{\mu^2-1}} \tag{21}$$

$$\tan(\alpha_m - \alpha_k) = \frac{\tan\alpha_m - \tan\alpha_k}{1 + \tan\alpha_m \tan\alpha_k} \tag{22}$$

From (20,21,22), we can express $\tan\alpha_k$ with respect to the number of pixels k as :

$$\tan\alpha_k = \frac{k\sqrt{\mu^2-1}}{m\mu^2 - k}. \tag{23}$$

From (14,17,23), we obtain :

$$2\sin^2\left(\frac{\beta_k}{2}\right) = (\sqrt{\mu^2-1} - \sin\beta_k)\frac{k\sqrt{\mu^2-1}}{m\mu^2-k}, \tag{24}$$

for $k = 1, \ldots, 2m$. As can be seen in (24), we derive the angle β_k as an implicit function of μ. After the evaluation of the angle β_k for $k = 2, \ldots, 2m$, we can derive the size of a cross-section arc corresponding to the projection of a line segment of constant size in the image :

$$|\widehat{A_kA}_{k-1}| = (\beta_k - \beta_{k-1})\, r. \tag{25}$$

We normalize the arc length $|\widehat{A_kA}_{k-1}|$ by the arc $|\widehat{A_mA}_{m-1}|$ corresponding to the image region situated at the intersection with the focal axis. The normalized arc lengths $\frac{|\widehat{A_kA}_{k-1}|}{|\widehat{A_mA}_{m-1}|}$, calculated based on (25) for $m = 100$ when $\mu \in \{1.5, 3, 25\}$ are represented in Figure 6. From this plot, we observe that arcs of different lengths on the cross-section are projected to segments with the same length in the image plane. This creates a nonlinear distortion in the image when compared to the original scene on the SUGCC. The distortion becomes visible because of the loss in resolution when projecting the scene and because of the change in light incidence angle [4, 24]. The angle β_k lies in the range $\left[0, \arccos\left(\frac{r}{l}\right)\right]$. The largest distortion occurs in the regions situated near the contact points of the tangents from the viewpoint to the cylindrical surface (limb points), where β_k is close to 0. The symmetry of the plot shown in Figure 6 can be easily verified from (24). The region where the curves bottom out can be considered as having small geometrical distortions.

3.2 Estimating the necessary number of views for image mosaicing

It was proven in the previous Section that distortions occur when projecting a scene from a SUGCC to an image plane. These distortions depend on the ratio μ (18), as shown in Figure 6. The region from the SUGCC which projects in the image without significant distortions contains arcs having a small size variation, while increasing β_k :

$$\left| \frac{|\widehat{A_k A}_{k-1}| - |\widehat{A_{k-1} A}_{k-2}|}{|\widehat{A_m A}_{m-1}| - |\widehat{A_{m-1} A}_{m-2}|} \right| = \left| \frac{\beta_k + \beta_{k-2} - 2\beta_{k-1}}{\beta_m + \beta_{m-2} - 2\beta_{m-1}} \right| \leq \delta, \tag{26}$$

where $|\widehat{A_k A}_{k-1}|$ and $|\widehat{A_{k-1} A}_{k-2}|$ are evaluated in (25), and δ is a small constant measuring the difference in the arc length variation and represents a distortion measure. Let us suppose that the minimal distortion condition (26) is satisfied for $k = d, \ldots, 2m - d$, where d is the pixel index for which the equality in (26) holds and corresponds to the number of pixels from one of the two image regions with big distortions.

When mosaicing the images obtained from various viewpoints it is assumed that they contain overlapping areas. However, regions which contain large distortions are not appropriate to be used for mosaicing. The distortion in the image depends on the distance from the intersection with the focal axis, as can be seen in Figure 6, where we have delimited the region with small geometrical distortions (26) by using two vertical arrows. The overlapping regions of two images taken from different viewpoints are likely to be distorted differently. Distortions or matching failure may arise when trying to mosaic the two images having uneven geometrical distortions in their overlapping region. In Figure 1 three images taken from three different viewpoints located around a cylinder and having overlapping regions are presented.

Let us assume that the viewpoints for the images to be mosaiced are located at the same distance l from the SUGCC axis. This can be obtained by constructing virtual images as described in Section 2.3. The size of the overlapping region should be larger than d, where d is obtained from (26). The angle corresponding to the arc $|\widehat{A_1 A_d}|$ is denoted by β_d. This arc corresponds to half of the minimum necessary overlapping region size in order to perform mosaicing. The image region containing small geometrical distortions and its corresponding part from the SUGCC surface are represented with thick line in Figure 7. The image size that is unreliable for mosaicing because of the perspective projection, evaluated from (24)-(26), is represented in Figure 8, when considering $m = 1000$. As can be observed from this plot, when μ increases, the size of the region containing geometrical distortions decreases.

After evaluating the minimal overlapping region size, which must be larger than the distorted area of the flattened SUGCC, we can obtain the necessary number of views to be taken around a SUGCC in order to represent the entire scene. The angle from the SUGCC axis corresponding to the overlapping

region (overlapping angle) must be larger than $2\beta_d$ in order to appropriately match the two regions. This condition provides a minimal bound for the number of images. If we consider that each pixel in the scene should be projected in two neighboring images at most, we obtain an upper limit for the necessary number of images. Thus the number of images necessary to represent the entire scene depicted on the SUGCC surface lies in the interval :

$$\frac{\pi}{\arccos\left(\frac{r}{l}\right) - \beta_d} < n < \frac{2\pi}{\arccos\left(\frac{r}{l}\right)}, \tag{27}$$

where the angle β_d is derived from (26). The following condition must hold in order for (27) to be true :

$$\cos 2\beta_d < \frac{r}{l}. \tag{28}$$

which corresponds to :

$$\beta_d < \frac{\beta_m}{2} \tag{29}$$

with $\widehat{A_1 O' O} = \beta_m$ in Figure 5. This means that more than half of the SUGCC's surface is projected on the image with minimal distortions, which is true if δ is small enough. The minimal and maximal bounds for the necessary number of views which should be taken around a SUGCC in order to represent the entire scene without distortion are shown in Figure 9. As can be seen from this plot, the minimal necessary number of views decreases and converges towards two when the distance from the viewpoint to the revolution axis increases.

4 Mosaicing the flattened SUGC surfaces

In Section 2 we have shown how we can find the localization parameters for a SUGC. By flattening, the 3-D view is transformed in a 2-D representation and the mosaicing problem can be treated in 2-D. In this Section we present a new 2-D mosaicing approach in order to reconstruct the entire scene. The proposed mosaicing algorithm is applied on images representing flattened surfaces. After reconstructing the entire scene we show how we can map the image on a virtual surface.

4.1 Flattening the curved surface

Based on the parameters derived in Section 2, we can flatten the SUGC surface. Let us consider a SUGC with $z = f(y)$, $\forall x$, where $f(y)$ is the equation describing a cross-section of the surface.

For two points P_k and P_{k-1} on the SUGC, we take two new points P'_k and P'_{k-1}, on the plane $z = z_m$, $\forall x, y$ such that :

$$\begin{aligned} |\widehat{P_k P}_{k-1}| &= \sqrt{(y_k - y_{k-1})^2 + (z_k - z_{k-1})^2} = \\ = |P'_k P'_{k-1}| &= |y'_k - y'_{k-1}|, \end{aligned} \tag{30}$$

where y'_k is the y coordinate of P'_k, $k \in \{1, \ldots, 2m\}$ and $|\cdot|$ is the absolute value. In fact, for a plane we have $z'_k = z_m$, where z'_k is the coordinate of P'_k on z axis. Thus, we derive :

$$y'_{m+k} = y'_{m+k-1} + |\widehat{P_{m+k} P}_{m+k-1}| = y'_m \pm \sum_{n=m}^{n=m+k} |\widehat{P_n P}_{n-1}|, \tag{31}$$

with $k \in \{0, \ldots, m\}$. We consider P'_m fixed, $y'_m = y_m$ and we calculate the coordinates y_k of the other points P'_k as illustrated in Figure 10. Thus, for each point P_k of the SUGC, we obtain a new point P'_k belonging to the flattened surface. Finally, from the perspective view of an SUGC, we derive an image where the curves have been straightened. The flattening of the paintings situated on an arch illustrated in Figures 2 a and b are shown in Figures 11 a and b. We can observe in these Figures that the two cross-sections representing the painting borders are not perfectly straightened. The most important source of error is the small misfit of the assumed equation for the SUGC compared to the real surface equation.

4.2 Image mosaicing

Image mosaics are created from a set of overlapping images in order to obtain a complete image scene with fine details [8]. Mosaicing consists in finding the relative positions of the images in the final scene description.

Let us denote by $\mathbf{W}_{p-1}$, the vector representing the coordinates of a pixel from the p-th image to be mosaiced. The relative positions of two images can be found from the following relationship :

$$\mathbf{W}_p = K \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \mathbf{W}_{p-1} + \mathbf{D} \tag{32}$$

where $\mathbf{D}$ is the displacement vector, θ is a rotation angle and K is the scaling factor. Several images can be mosaiced after arranging them two by two. A mosaicing algorithm has to find all these parameters. In this paper we propose an image mosaicing algorithm based on matching overlapping regions. The images are taken from various viewpoints as shown in Figure 1. After flattening the images from the SUGC as described in Section 4.1, we match the corresponding regions from each two neighboring images. The neighboring images must contain overlapping regions in order to find their relative positions. Similarly to the block matching algorithms [14], we define a search area $S_u \times S_v$ in the plane uOv. This search

region includes the overlapping part of the two images. We search for the best matching between various regions from the two images in the given area :

$$\min_{i,j,\theta_l} \left| I(\mathbf{W}_p) - I\left(K \begin{bmatrix} \cos\theta_l & -\sin\theta_l \\ \sin\theta_l & \cos\theta_l \end{bmatrix} \mathbf{W}_{p-1} + \mathbf{D}_{ij} \right) \right| \tag{33}$$

where we consider only the pixels from the resulting overlapping region and where $\mathbf{D}_{ij}$ is the displacement assumed in a certain range $\mathbf{D}_{ij} \in S_u \times S_v$ and defined at the margin of the flattened region, $\theta_l \in (-\pi, \pi]$ is a rotation angle and $I(\cdot)$ is the graylevel value. The matching procedure is automatic, the only necessary parameters are those of the search region S_u and S_v. The procedure of matching is illustrated in Figure 12 a. In Figure 12 b we show the matching result of the two flattened painted arch images. The original arch images are shown in Figures 2 a and b.

When considering the matching of flattened surfaces we must take into account the distortions which arise due to perspective projection. The distortion analysis was provided in Section 3.1. The matching between the overlapping areas of two neighboring images according to (33) may fail, due to perspective distortions resulting after flattening. The size d of the distorted area from the image, calculated according to (26), is shown in Figure 8. The regions which contain geometrical distortions are excluded from the matching procedure (33). Therefore, the search region should be larger than the region of distortion in the flattened image :

$$S_u > 2d. \tag{34}$$

After mosaicing according to (33), the pixel values from the common parts are calculated based on a weighting function, similar to that used in [10]. First, we evaluate the minimum distance from a given pixel site (i, j) to the border of the common reliable part (which does not contain geometrical distortions due to perspective projection) :

$$g_{p-1} = \min\{i, j\} \ , \ g_p = \min\{D_i - i, D_j - j\} \tag{35}$$

for $i = 1, \ldots, S_u$ and $j = 1, \ldots, S_v$, where $\mathbf{D}_{ij} = (D_i, D_j)$ denotes the displacement between the two images as provided by (33). The pixels of the resulting mosaic are obtained by weighting the pixel values of the two original images :

$$I(i,j) = \frac{g_p}{g_p + g_{p-1}} I_p(i,j) + \frac{g_{p-1}}{g_p + g_{p-1}} I_{p-1}(i,j), \tag{36}$$

where $I(i,j)$ and $I_p(i,j)$ denote the graylevel values of the pixels from the mosaiced image and from the p-th image, respectively. This procedure ensures a smooth luminosity transition between each two images, as shown in Figure 12 b. In the case when the images to be mosaiced contain exactly the same

graylevel in the respective pixel, this value is assigned to the final image pixel. This procedure can easily be extended for mosaicing multiple images, overlapping on both u and v axes.

4.3 Mapping of the reconstructed scene

Let us consider known the function of a certain surface where we want to map the reconstructed image. Such a surface can be a cylindrical surface as represented in Figure 13. The horizontal size of the mosaiced image is considered equal to M pixels. Let us denote by P_k, $k \in \{1, \ldots, M\}$ the pixels located on the horizontal cross-section through the SUGCC.

Let us denote by Δ, the distance between two pixels. The mapping is performed by maintaining a constant distance Δ between each two consecutive pixels. The step Δ on the cylinder, is defined by :

$$\Delta = \frac{1}{2\pi r}, \tag{37}$$

where r is the radius of the cylinder.

We obtain the coordinates y_k and z_k for each point P_k. In parametric coordinates we have the relations :

$$\begin{cases} y_k &= r\cos(2\pi k\Delta) \\ z_k &= r\sin(2\pi k\Delta) \end{cases}, \tag{38}$$

where Δ is the step defined by (37) and $k \in \{1, \ldots, M\}$. In order to avoid image wrapping at its extremities, when mapped on the cylinder, we impose the following condition for the radius :

$$r > \frac{2M}{\pi}. \tag{39}$$

where M is the image size on the v axis. After mapping the reconstructed scene on this surface, we can create synthetic views from any viewpoint.

5 Simulation Results

In Figure 1, the projection and scene reconstruction from the surface of a cylinder is illustrated. From the mosaicing point of view, a similar problem is also when we try to mosaic the surface of a rotating SUGC, while the stationary camera takes shots of its surface at certain time intervals (correlated with the SUGC's rotation speed). The proposed flattening algorithm has been used on several examples of SUGC's images. The mosaicing algorithm has been tested on various sets of flat or curved surface images. In the following, we present the application of the proposed combined flattening and mosaicing algorithm

when used for reconstructing the scene from the surface of a cup. This object contains two parallels all around its surface, delimiting the decorative pattern. The parallels are used to find the localization parameters, when taking the pictures from different viewpoints, as it was described in Section 2.

Three images are taken from different viewpoints such that each two of them have common regions as shown in Figures 14 a, b, c. In these Figures we can observe the geometrical distortions caused by the perspective projection at the margins of the cup views. The internal parameters of the camera, i.e., the focal distance f and the resolution factor s, used in this experiment are provided in Table 1. From Table 1 and from (18) we calculate that $\mu = 5.5$. We can observe from Figure 9 that we need four images in order to represent the entire surface. Due to the cup handle and because the scene does not cover the entire cup surface, we use only three images for reconstructing the scene in our experiments. The image size is $(2u_0, 2v_0) = (288, 384)$ pixels. The localization parameters, which have been found as described in Section 2.3, are provided in Table 2. For each image, we have used two parallels in order to find the rotation angles. The parallels corresponding to the image from Figure 14 b are shown in Figure 14 d. In Figure 14 e is shown the common normal of the two parallels, corresponding to the projection of the revolution axis. Using the common normal, after evaluating the position of the second axis as described in Section 2.2, and after rotating the image from Figure 14 b, we obtain the virtual image in Figure 14 f.

Based on the localization parameters from Table 2, we rotate the camera for each image and we obtain a set of virtual images as described in Section 2.3. The overlapping parts of each two images representing neighboring regions from the cup surface fulfill the minimal size condition according to (34) and to the plot from Figure 8. The three images are flattened as described in Section 4.1. The flattened images after appropriate scaling are shown in Figures 14 g, h, i. The regions located at the margins of the pictures contain certain distortions produced by the perspective distortion as it can be observed in the decorative pattern. Based on the analysis provided in Section 3 we find the reliable regions for matching. Matching reliable image regions, i.e. the ones produced after clipping the regions with big distortions, is shown in Figures 14 j, k, l. The regions with big perspective distortions can be eliminated either before or after flattening, and they correspond to the angle β_d from the cylinder axis and to d pixels in the flattened image, respectively. However, by eliminating this region before flattening we reduce the computational time for the matching algorithm. The mosaicing of the three pictures is provided in Figure 15. This Figure represents the flattened version of the entire scene depicted on the cup.

After mosaicing the flattened pictures, we map the scene on a virtual cylinder as described in Section 4.3. Finally, we simulate a camera to obtain a new image, from a different viewpoint, as shown in Figure 16.

6 Conclusion

In this study we propose a new approach for mosaicing a scene mapped or painted on a SUGC. We estimate the localization parameters of SUGC's surfaces in images. The localization used the projection of the SUGC's axis and the second axis perpendicular on it. Based on the localization parameters, each image is backprojected on the SUGC's surface in order to be flattened. We provide a theoretical analysis of the geometrical distortions due to the perspective projection. Using this analysis we derive the necessary number of views in order to represent the scene depicted on a SUGCC. We apply a matching algorithm in order to estimate the displacements of the flattened surfaces with respect to each other. The displacement parameters are used for mosaicing the entire scene resulting from the overlapping images. This approach was successfully applied to real examples. In the first example shown in this study we have reconstructed the scene painted on a church arch and in the second one we have reconstructed and visualized the entire ornamental scene depicted on a cup.

This work can be applied for the visualization of mural paintings when they are located on SUGC's surfaces. In this case, we can obtain a better visualization and description of the painting. The respective painting can be viewed from various viewpoints while mapped on a virtual surface. This information can be further used for painting restoration or for finding the optimal observation viewpoint for the respective mural painting.

References

[1] M. Dhome, M. Richetin, J.-T. Lapresté, and G. Rives, Determination of the Attitude of 3D Objects from a Single Perspective View, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. **11**, no. 12, pp. 1265-1278, (1989).

[2] M. Richetin, M. Dhome, J.-T. Lapresté, and G. Rives, Inverse Perspective Transform Using Zero-Curvature Contour Points: Application to the Localization of Some Generalized Cylinders from a Single view, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. **13**, no.2, pp. 185-192, (1991).

[3] Y.C. You, J.D. Lee, J.Y. Lee, and C.H. Chen, Determining Location and Orientation of a Labeled Cylinder Using Point-Pair Estimation Algorithm, *Proc. 11th International Conference on Pattern Recognition*, Hague, Netherlands, vol. **I**, pp. 354-357, (1992).

[4] O. Wink, A.W.M. Smeulders, and D.C. Koelma, Location Estimation of Cylinders From a 2-D Image, *Proc. 12th International Conference on Pattern Recognition*, Jerusalem, Israel, I, pp. 682-684, (1994).

[5] D.P. Huttenlocher, and S. Ullmann, Recognizing Solid Objects by Aligning with an Image, *International Journal of Computer Vision*, vol. **5**, no. 2, pp. 195-212, (1990).

[6] H. Tanahashi, K. Sakaue, and K. Yamamoto, Recovering Decorative Patterns of Ceramics Objects from Monocular Image Using a Genetic Algorithm, *Proc. 3rd International Conference on Document Analysis and Recognition*, Montréal, Canada, vol. I, pp. 339-342, (1995).

[7] B. J. Super, A. C. Bovik, Shape from texture using local spectral moments, *IEEE Trans. on PAMI,* vol. **17**, no. 4, pp. 333-343, (1995).

[8] R.J. Schalkoff, Digital Image Processing and Computer Vision. *John Wiley:* New York, (1989).

[9] R.J. Billinge, J. Cupitt, J. Dessipiris, and D. Saunders A Note on an Improved Procedure for the Rapid Assembly of Infrared Reflectogram Mosaics, *Studies in Conservation*, vol. **38**, pp 92-97, (1993).

[10] P. Jaillon, and A. Montanvert, Image Mosaicking Applied On 3D Surfaces, *In Proc. 12th International Conference on Pattern Recognition*, Jerusalem, Israel, vol. **I**, pp. 253-257, (1994).

[11] R. Szelinski, Video Mosaics for Virtual Environments, *IEEE Computer Graphics and Applications,* vol. 16, no.2, pp. 22-30, (1996).

[12] Q. Zheng, R. Chellappa, A computational Vision Approach to image Registration, *IEEE Trans. on Image Processing,* vol. **2**, no. 3, pp. 311-326, (1993).

[13] Ş. Gümüştekin, R. W. Hall, Mosaic Image Generation on a Flattened Gaussian Sphere, *Proc. IEEE Workshop on Applications of Computer Vision,* Sarasota, USA, pp. 50-55, (1996).

[14] A.N. Netravali, and B.G. Haskell, Digital Pictures, Representation and Compression. *Plenum Press*, (1988).

[15] A. G. Borş, W. Puech, I. Pitas, J.-M. Chassery, "Perspective Distortion Analysis for Mosaicing Images Painted on Cylindrical Surfaces," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97),* Munich, Germany, vol. **4**, pp. 3049-3052, (1997).

[16] J.-M. Chassery, Image Processing And Analysis. A Challenge For Art Paintings, *Proc. Art and Technology Workshop*, Athens, Greece, pp. 366-377, (1993).

[17] O.D. Faugeras, Three-Dimensional Computer Vision: A Geometric Viewpoint. *MIT Press*, (1993).

[18] W. Puech, P. Schott, and J.-M. Chassery, Discrete Method of Calculus Using Common Normals for Curved Surface Reconstruction, *In Proc. 5th Discrete Geometry for Computer Imagery*, Clermont-Ferrand, France, pp. 83-92, (1995).

[19] W. Puech, and J.-M. Chassery, Curved Surface Reconstruction Using Monocular Vision, *Proc. 8th European Signal Processing Conference*, Trieste, Italy, pp. 9-12, (1996).

[20] M. Brady, Criteria for Representations of Shape. *Human and Machine Vision, A. Rosenfeld and J. Beck*, New York, (1983).

[21] J. Ponce, D. Chelberg, and W. Mann, Invariant Properties of Straight Homogeneous Generalized Cylinders and their Contours, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. **11**, no. 9, pp. 951-966, (1989).

[22] R. Glachet, M. Dhome, and J.-T. Lapresté, Finding the Perspective Projection of an Axis of Revolution, *Pattern Recognition Letters*, vol. **12**, no. 11, pp. 693-700, (1991).

[23] W. Puech, J.-M. Chassery, and I. Pitas, Cylindrical Surface Localization in Monocular Vision, *Pattern Recognition Letters*, vol. **18**, pp. 711-722, (1997).

[24] M. Oren, and S.K. Nayar, Generalization of the Lambertian Model and Implications for Machine Vision, *International Journal of Computer Vision*, vol. **14**, no. 3, pp. 227-251, (1995).

focal distance	resolution factor	Radius	distance
100 mm	10 pixels/mm	100 mm	550 mm

Table 1: The given parameters for experiments.

	View 1	View 2	View 3
θ_x	36.0°	42.0°	36.0°
θ_y	0.10°	-2.19°	1.37°
θ_z	3.84°	3.39°	2.94°

Table 2: The rotation angles found for the three views of a cup.

List of Figures

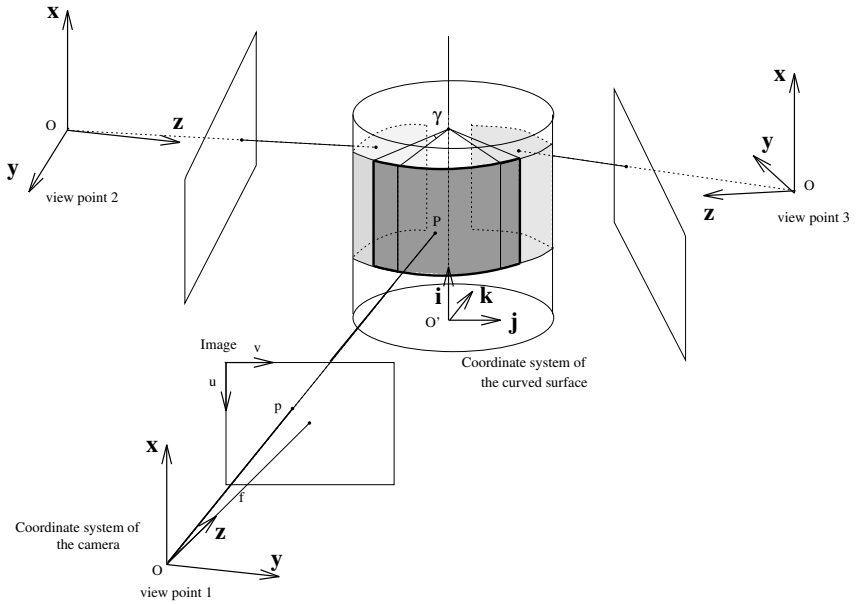


Figure 1: Images taken from different viewpoints located around a SUGC.

(a) (b)

Figure 2: Left and right views of a painted church arch.

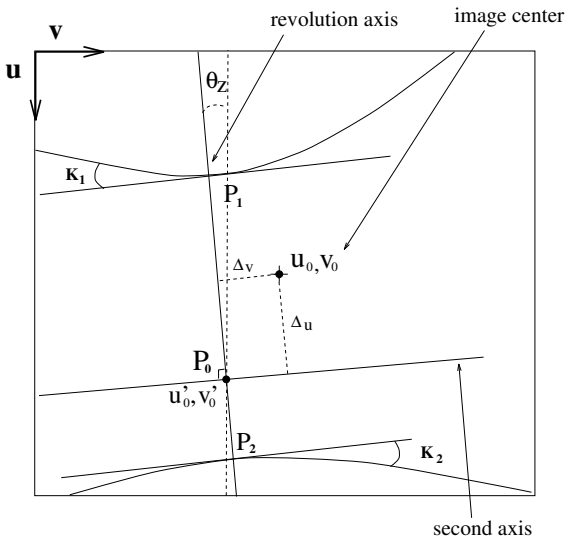


Figure 3: The SUGC axis and the second axis detected from two cross-section projections.

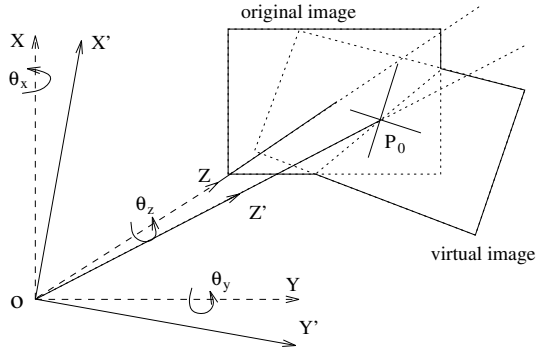


Figure 4: A virtual image obtained after camera rotation.

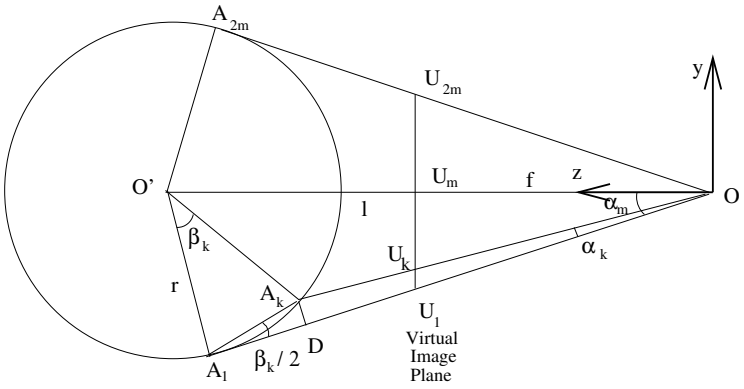


Figure 5: The cross-section representation through the SUGCC and the virtual image plane.

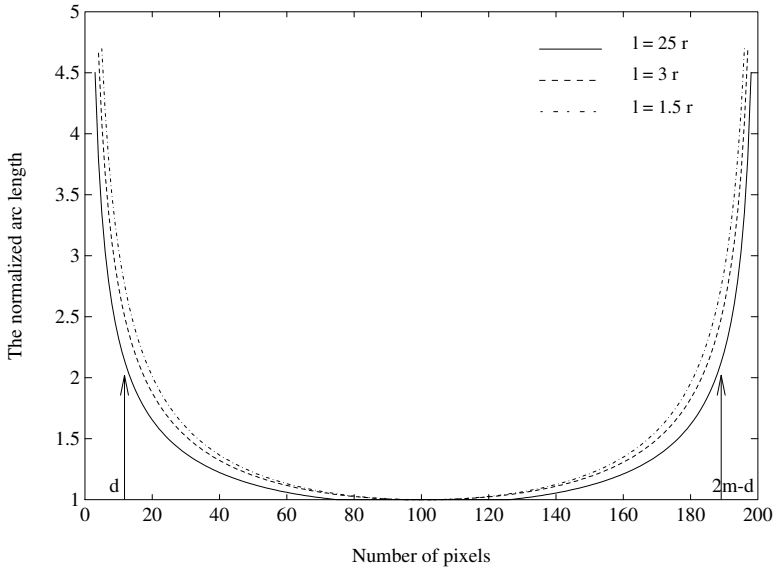


Figure 6: The arc length variation where each arc corresponds to an equally sized segment in the image.

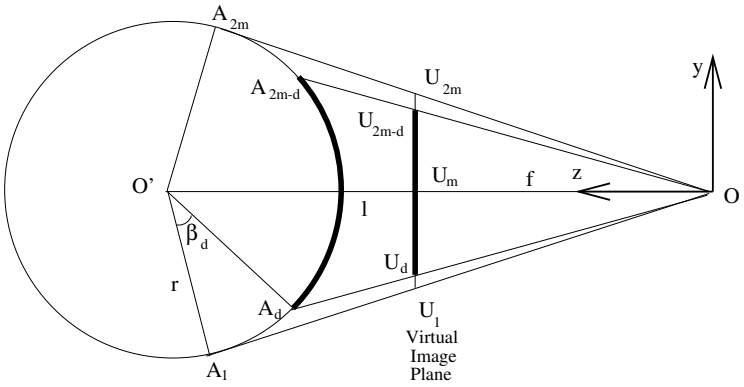


Figure 7: Cross-section representation through SUGCC and virtual image plane. The image region with reliable geometrical distortions is drawn with thick line.

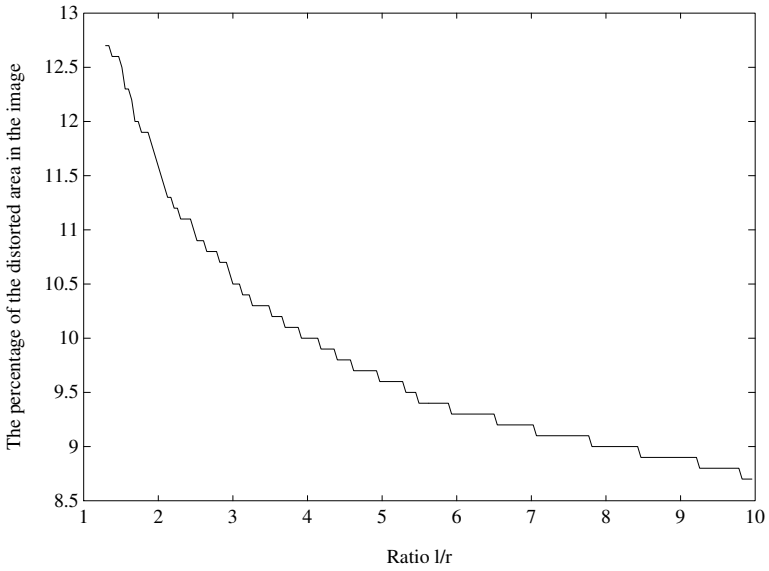


Figure 8: Evaluation of the distortion caused by the perspective projection.

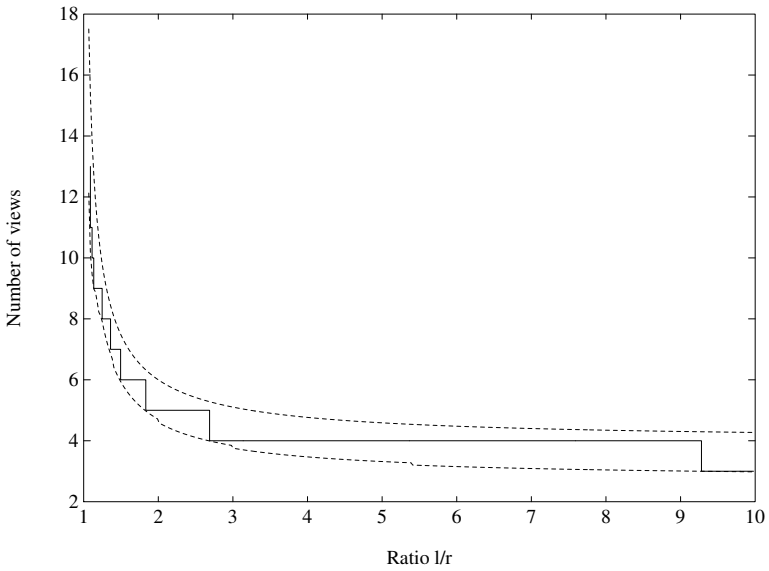


Figure 9: The necessary number of views to represent the entire scene from a SUGCC function of the ratio l/r.

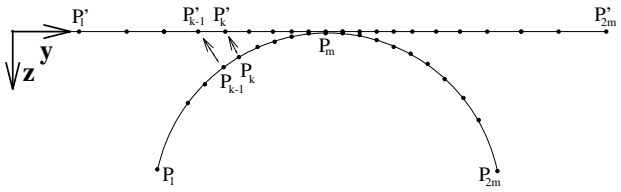
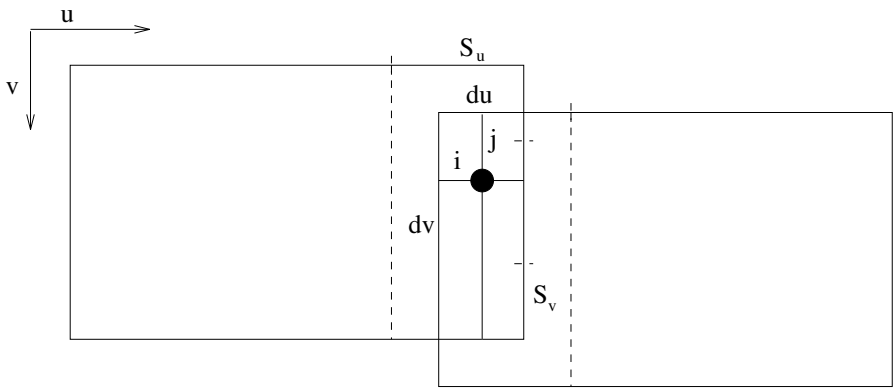


Figure 10: A cross-section illustrating the flattening of the SUGC surface.

(a) (b)

Figure 11: Flattened paintings after localization.



(a)

(b)

Figure 12: The matching between two images with a common region. (a) The illustration of the approach. (b) The matching obtained when applied on a painting, located on an arch, after being flattened.

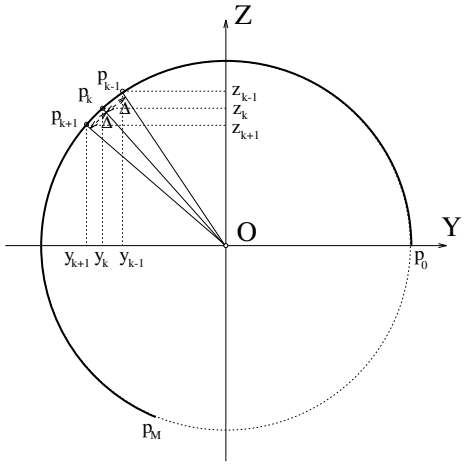


Figure 13: Mapping the image on a cylinder.

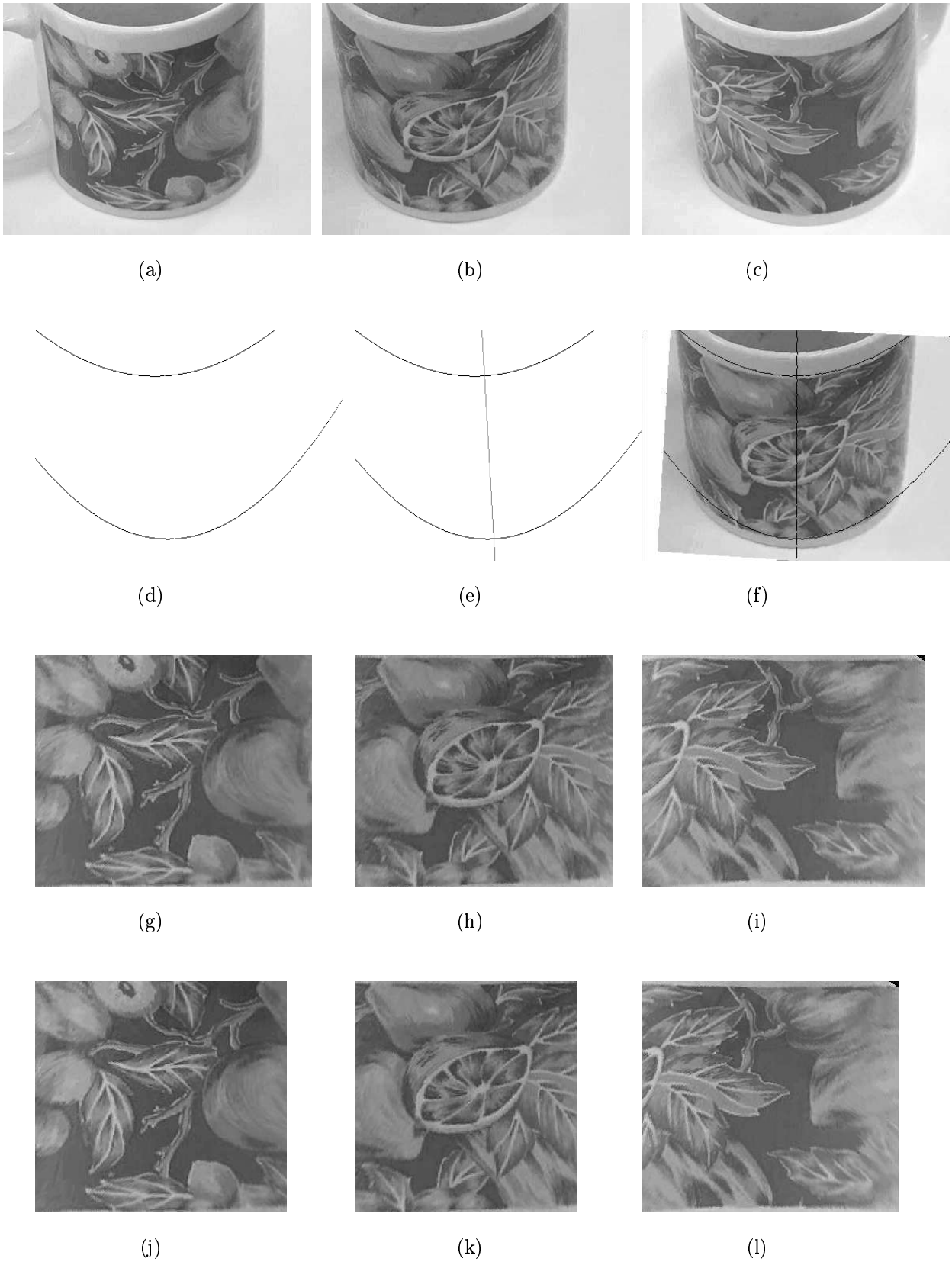


Figure 14: (a), (b), (c) The original images representing the cup surface, taken from three different viewpoints ; (d) The two cross-sections used to find the three rotation angles for (b); (e) The common normal corresponding to the cross-sections; (f) A virtual image after two rotations (θ_x, θ_z) when applied on the image from (b); (g), (h), (i) The flattened surface representations of the three images; (j), (k), (l) The flattened surface representations of the three images where the regions with large geometrical distorsion are clipped off.

Figure 15: The result of the mosaicing algorithm representing the entire decorative scene.

Figure 16: Synthetic visualization of the scene, mapped on a virtual cylinder.