Multiperspective Projection and Collineation

Jingyi Yu
Massachusetts Institute of Technology
Cambridge, MA 02139
jingyiyu@csail.mit.edu

Leonard McMillan
University of North Carolina
Chapel Hill, NC 27599
mcmillan@cs.unc.edu

Abstract

We present theories of multiperspective projection and collineation. Given an arbitrary multiperspective imaging system that captures smoothly varying set of rays, we show how to map the rays onto a 2D ray manifold embedded in a 4D linear vector space. The characteristics of this imaging system, such as its projection, collineation, and image distortions can be analyzed by studying the 2-D tangent planes of this ray manifold. These tangent planes correspond to the recently proposed General Linear Camera (GLC) model.

In this paper, we study the imaging process of the GLCs. We show the GLC imaging process can be broken down into two separate stages: the mapping of 3D geometry to rays and the sampling of those rays over an image plane. We derive a closed-form solution to projecting 3D points in a scene to rays in a GLC. A GLC image is created by sampling these rays over an image plane. We develop a notion of GLC collineation analogous to pinhole cameras. GLC collineation describes the transformation between the images of a single GLC due to changes in sampling and image plane selection. We show that general GLC collineations can be characterized by a quartic (4th order) rational function. GLC projection and collineation provides a basis for developing new computer vision algorithms suitable for analyzing a wider range of imaging systems than current methods, based on simple pinhole projection models, permit.

1. Introduction

The imaging process entails mapping 3D geometry onto a two dimensional manifold via some camera or imaging model. This projection depends on both the geometry of the imaging system and the parametrization of the image plane. The most common imaging model is the pinhole camera, which collects rays passing through a point and organizes rays onto a image plane. This mapping can be fully described using the classic 3 x 4 camera matrix [5].
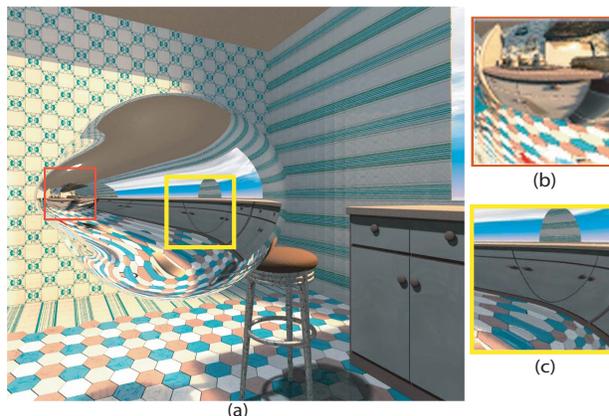
Figure 1. (a) An image captured on a pear-shaped mirror. (b) Cross-slit type distortions. (c) Pushbroom type distortions.

These mappings are unique down to a scale factor, and the same infrastructure can also be used to describe orthographic cameras. When the image plane of the pinhole camera changes, the corresponding images change according. The transformation between images on two different planes is often referred to as homography. In the pinhole case, it is a projective transformation and, thus, lines are still preserved as lines despite the change of the image plane.

Recent developments have suggested alternative imaging models such as pushbroom [10], cross-slit [16, 8], and oblique [7] cameras. These cameras collect rays under different geometric constraints. For instance, all rays pass through a particular line in a pushbroom camera. All rays of a cross-slit cameras pass through two lines, and, in oblique cameras no two rays can intersect or be parallel. These camera models are often referred to as multiperspective cameras. They provide alternate and potentially advantageous imaging systems for understanding the structure of observed scenes. The mapping from 3D points to pixels is no longer a projective transformation in these cameras, and is often difficult to calculate. For instance, the collineation is a 3x3x3 tensor in a cross-slit camera [16]. Interesting multiperspective distortions are also observed on these cam-

eras, as shown in Figure 1.

When we model complex imaging systems such as cata-dioptric mirrors, it is difficult to compute the mapping from 3D points to 2D images because it requires an inverse mapping from rays to points and a closed-form solution may not exist. The lack of a closed-form projection prohibits further analysis of the resulting images. In practice, however, many catadioptric imaging systems do exhibit local distortions that are similar to pushbroom or cross-slit distortions (Figure 1). In this paper, we show that these visual phenomenon are not coincidental. In fact, these imaging systems can be precisely modelled by a special class of local multiperspective camera models. Furthermore, we give a closed-form solution for finding the local multiperspective camera models and demonstrate that all possible images can be formed by these cameras.

Given any imaging system, we first map the rays collected by the system to a two dimensional ray manifold embedded in a 4D ray space. We benefit from the recently proposed General Linear Camera (GLC) model [17], which describes all possible linear manifolds. We use the GLC model to locally analyze the ray manifold using its tangent plane which can be completely characterized as one of the eight GLCs. We provide a closed-form solution to Projection for all GLCs. Next, we show how to determine the Collineation using the GLC intrinsics. We show the Collineation between two image planes is, in general, a quartic (4th order) rational transformation. Most imaging systems, such as the catadioptric mirrors, can be easily analyzed using our framework and distortions can be interpreted using the local GLC Projection and Collineation model.

2. Previous Work

Pinhole cameras collect rays passing through a single point. Because of its simplicity, the projection is the classic 3 x 4 camera matrix [5], which combines six extrinsic and five intrinsic camera parameters into a single operator that maps homogenous 3D points to a 2D image plane. These mappings are unique down to a scale factor, and the same infrastructure can also be used to describe orthographic cameras. When only the image plane changes while the pinhole remains constant, the transformation between the two images is a projective transformation, which is called a homography.

Recently, several researchers have proposed alternative multiperspective camera models, which capture rays originating from different points in space. These multiperspective cameras include pushbroom cameras [10], which collect rays along parallel planes from points swept along a linear trajectory, two-slit cameras [8, 16], which collect all rays passing through two lines, and oblique cameras [7],
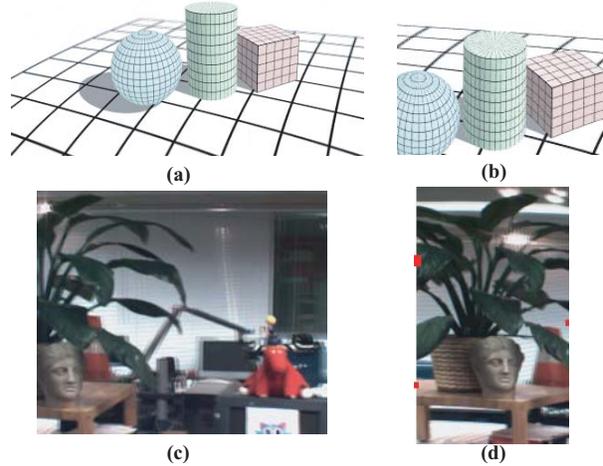

Figure 2. (a) and (c) are cross-slits images synthesized from light fields. (b) and (d) are pushbroom images synthesized from light fields. The distortions of the curved isolines on the objects illustrate various multi-perspective effects of GLC cameras in (a) and (b). The pushbroom camera (b) and (d) exhibits apparent stretching/duplications of objects lying far away from the image plane.

in which each pair of rays are oblique. The resulting images captured by these cameras are easily interpretable, yet they exhibit interesting multiperspective distortions. For instance, lines often project to curves as shown in Figure 2 and a single 3D point might project to multiple points, causing feature duplications shown in Figure 2(d).

Multiperspective imaging techniques have also been explored in the field of computer graphics. Example images include multiple-center-of-projection images [2], manifold mosaics [9], and multiperspective panoramas [15]. Multiperspective distortions have also been studied on real catadioptric imaging systems [1, 13] and are often characterized using caustics [13]. Zorin and Barr [19] studied the use of multiperspective and other geometric distortions to improve perceptual qualities of images. Swaminathan et al [14] proposed a method to compute minimally distorted images using simple geometry priors on scene structure.

Seitz [11] has analyzed the space of multiperspective cameras to determine those with a consistent epipolar geometry. His work suggests that only a small class of multiperspective images can be used to analyze three-dimensional structures, mainly because the epipolar constraints cannot be established over the these images.

General projection and imaging models have also been studied in terms of rays. Pajdla proposed the ray-closure model to describe a camera and its projection. Gu et al [4] explicitly parameterized rays under a particular 4D mapping known as a two-plane parametrization. Most recently, we characterize all linear manifolds in the 4D ray space defined by a two-plane parametrization, which we call Gen-
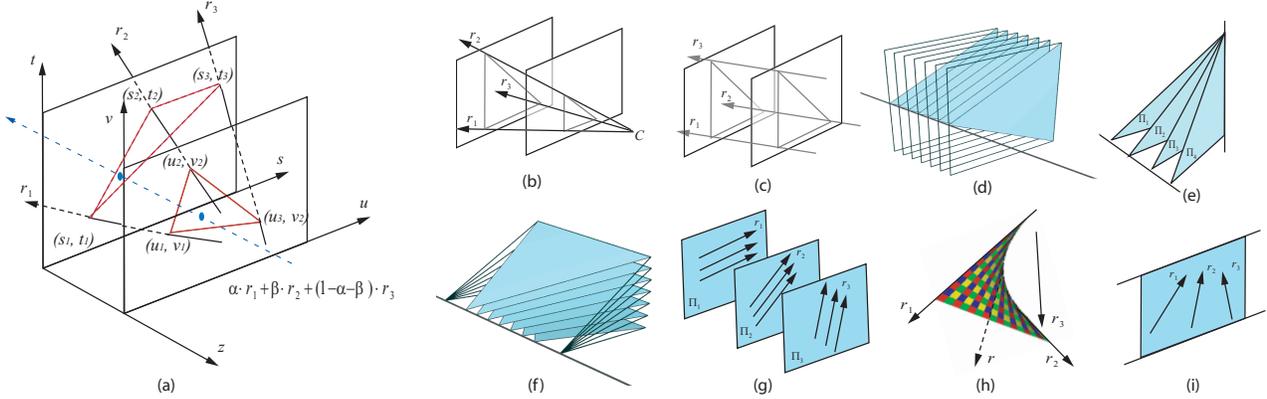
Figure 3. General Linear Camera Models. (a) A GLC Model collects radiance along all possible affine combination of three rays. The rays are parameterized by their intersections with two parallel planes. There are precisely eight GLCs, shown in (b) to (i). (b) In a pinhole camera, all rays pass through a single point. (c) In an orthographic camera, all rays are parallel. (d) In a pushbroom, all rays lie on a set of parallel planes and pass through a line. (e) In a cross slit camera, all rays pass through two non-coplanar lines. (f) In a pencil camera, all coplanar rays originate from a point on a line and lie on a specific plane through the line. (g) In a twisted orthographic camera, all rays lie on parallel twisted planes and no rays intersect. (h) In an bilinear camera, no two rays are coplanar and no two rays intersect. (i) In an EPI camera, all rays lie on a 2D plane.

eral Linear Cameras. Most of these cameras satisfy Seitz's criterion [11]. In this paper, we use GLCs as a first-order differential model for modelling arbitrary imaging systems.

3. General Linear Cameras

The general linear camera (GLC) model proposed unifies traditional perspective, orthographic, and multiperspective cameras models. In the GLC framework, every ray is parameterized by its intersections with the two parallel planes, where $[s, t]$ is the intersection with the first and $[u, v]$ the second, as shown in Figure 3(a). This parametrization is often called a two-plane parametrization (2PP) [6, 3]. Except for those rays parallel to the two planes, 2PP uniquely represents each ray by mapping it to a point in a four-dimensional ray space.

A GLC is defined as a plane in the 4D ray space and is specified as three rays as:

$$
\begin{aligned}
GLC \; = \; & \{r : r = \alpha \cdot [s_1, t_1, u_1, v_1] + \beta \cdot [s_2, t_2, u_2, v_2] \\
+ \; & (1 - \alpha - \beta) \cdot [s_3, t_3, u_3, v_3], \forall \alpha, \beta\}
\end{aligned}
$$

Most well-known multiperspective cameras, such as pushbroom, cross-slit, linear oblique cameras are GLCs. We simplify the analysis of [17] by substituting $\sigma = s - u$ and $\tau = t - v$. In this paper, we will use this $[\sigma, \tau, u, v]$ parametrization to represent rays. We also assume the default uv plane is at $z = 0$ and st plane at $z = 1$. To determine the type of the multiperspective camera, [17] provides a characteristic equation that computes how many slits

(lines) in 3D space that all rays will pass through:

$$
\begin{vmatrix}
u_1 + \lambda \cdot \sigma_1 & v_1 + \lambda \cdot \tau_1 & 1 \\
u_2 + \lambda \cdot \sigma_2 & v_2 + \lambda \cdot \tau_2 & 1 \\
u_3 + \lambda \cdot \sigma_3 & v_3 + \lambda \cdot \tau_3 & 1
\end{vmatrix} = 0 \qquad (1)
$$

which results in a quadratic equation of the form $A\lambda^2 + B\lambda + C = 0$ where

$$
A = \begin{vmatrix} \sigma_1 & \tau_1 & 1 \\ \sigma_2 & \tau_2 & 1 \\ \sigma_3 & \tau_3 & 1 \end{vmatrix}, \; C = \begin{vmatrix} u_1 & v_1 & 1 \\ u_2 & v_2 & 1 \\ u_3 & v_3 & 1 \end{vmatrix}
$$

$$
B = \begin{vmatrix} \sigma_1 & v_1 & 1 \\ \sigma_2 & v_2 & 1 \\ \sigma_3 & v_3 & 1 \end{vmatrix} - \begin{vmatrix} \tau_1 & u_1 & 1 \\ \tau_2 & u_2 & 1 \\ \tau_3 & u_3 & 1 \end{vmatrix} \qquad (2)
$$

The GLC type is determined by its A, B, and C coefficients and its discriminant $\Delta = B^2 - 4AC$ of the characteristic equation. A total of eight GLC types describe all 2D linear manifolds in the ray space, and their characteristics are shown in Table 1.

3.1. Local GLC Model

Given any imaging system that describes a continuous set of rays, e.g., catadioptric imaging systems or multiperspective panoramas, we can analyze this system using local GLC models. Specifically, let $\Sigma(x, y)$ be a continuous 2D ray manifold implicitly parameterized in x and y, i.e.,

$$
\Sigma(x, y) = [\sigma(x, y), \tau(x, y), u(x, y), v(x, y)] \qquad (3)
$$

We can approximate the local behavior of the rays by computing the tangent plane about any specified ray. The tangent plane can be expressed as two spanning vectors $\vec{d_1}$ and

$\vec{d_2}$:
$$\vec{d_1} = [\sigma_x, \tau_x, u_x, v_x], \vec{d_2} = [\sigma_y, \tau_y, u_y, v_y] \quad (4)$$

Note that every tangent plane corresponds to a GLC. Therefore, we can use the characteristic equation to determine the local GLC-type of every tangent plane. This corresponds to choosing three points on the tangent plane, $\Sigma(x,y)$, $\Sigma(x,y) + \vec{d_1}$, and $\Sigma(x,y) + \vec{d_2}$ using (2) as:

$$\begin{vmatrix} u + \lambda\sigma & v + \lambda\tau & 1 \\ (u + u_x) + \lambda(\sigma + \sigma_x) & (v + v_x) + \lambda(\tau + \tau_x) & 1 \\ (u + u_y) + \lambda(\sigma + \sigma_y) & (v + v_y) + \lambda(\tau + \tau_y) & 1 \end{vmatrix} = 0$$
$$(5)$$

yielding to the quadratic equation $A\lambda^2 + B\lambda + C = 0$ where

$$\begin{aligned} A &= \sigma_x\tau_y - \sigma_y\tau_x \\ B &= \sigma_x v_y - \sigma_y v_x - \tau_x u_y + \tau_y u_x \\ C &= u_x v_y - u_y v_x \end{aligned} \quad (6)$$

The discriminant, Δ, can be computed as

$$\begin{aligned} \Delta &= (\tau_y u_x - \tau_x u_y + \sigma_x v_y - \sigma_y v_x)^2 \\ &\quad - 4(\sigma_x\tau_y - \sigma_y\tau_x)(u_x v_y - u_y v_x) \end{aligned} \quad (7)$$

In Figure 1(a), we show the local GLC model for a peartop mirror surface. Our analysis from [18] shows that local GLC models on this mirror correspond to either crossslit 1(b) or pushbroom 1(c). In fact, the yellow region lie close to the parabolic curve of the surface.

4. GLC Projection

GLC projection is the mapping of points in 3D to to their corresponding ray in a specific GLC. We next consider projecting a 3D point onto a GLC. To simplify the analysis, we use a canonical GLC representations with the three generator rays $[\sigma_1, \tau_1, 0, 0]$, $[\sigma_2, \tau_2, 1, 0]$, and $[\sigma_3, \tau_3, 0, 1]$. This setup describe almost all GLCs (except for a subspace of GLCs whose slits lie on the uv plane). Every ray r in the GLC can be written as the following affine combination:

$$\begin{aligned} r[\sigma, \tau, u, v] &= (1 - \alpha - \beta) \cdot [\sigma_1, \tau_1, 0, 0] \\ &\quad + \alpha \cdot [\sigma_2, \tau_2, 1, 0] + \beta \cdot [\sigma_3, \tau_3, 0, 1] \end{aligned} \quad (8)$$

where σ_i, τ_i, i = 1, 2, 3 are constant for a given GLC. It is easy to see that $\alpha = u$ and $\beta = v$ under this simplification.
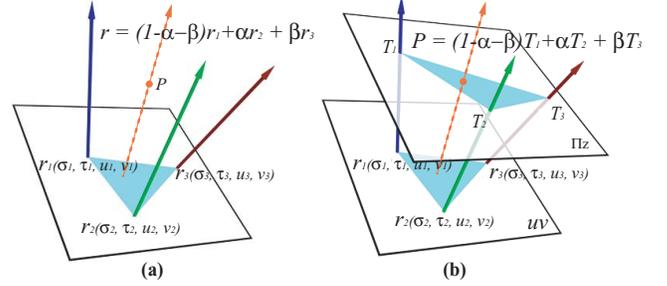
Figure 4. (a) Projecting a point P to a ray in the GLC. (b) The projection of P can be computed using the same affine coordinate on the sweeping plane Π_z.

Equation (8) is also equivalent to the following two linear constraints:

$$\begin{aligned} \sigma &= (1 - u - v)\sigma_1 + u\sigma_2 + v\sigma_3 \\ \tau &= (1 - u - v)\tau_1 + u\tau_2 + v\tau_3 \end{aligned} \quad (9)$$

The GLC ray that passes through a point $\dot{P}(x, y, z)$ in 3D satisfy the following linear constraints [4, 17]:

$$\begin{aligned} u + z \cdot \sigma &= x \\ v + z \cdot \tau &= y \end{aligned} \quad (10)$$

The ray passing through P is, thus, the solution of the four equations in (9) and (10) and can be computed as:

$$\begin{aligned} u &= -\frac{(z^2(\sigma_1\tau_3 - \sigma_3\tau_1) - z(\sigma_1(y-1) - \sigma_3 y - x(\tau_1 - \tau_3)) - x)}{Az^2 + Bz + C} \\ v &= \frac{(z^2(\sigma_1\tau_2 - \sigma_2\tau_1) - z(\sigma_1 y - \sigma_2 y + \tau_1(1-x) + \tau_2 x) + y)}{Az^2 + Bz + C} \end{aligned} \quad (11)$$

where $Az^2 + Bz + C = 0$ corresponds to the characteristic of the GLC. We call this equation the GLC Projection Equation.

4.1 Plane Sweeping

The GLC Projection Equation has an intuitive geometric interpretation. Consider a plane Π_z parallel to the uv plane and passing through $\dot{P}$. The three generators will intersect Π_z at $\dot{T}_1, \dot{T}_2, \dot{T}_3$, where

$$\begin{aligned} \dot{T}_1 &= (0,0,0) + z \cdot (\sigma_1, \tau_1, 1) = (\sigma_1 z, \tau_1 z, z) \quad (12) \\ \dot{T}_2 &= (1,0,0) + z \cdot (\sigma_2, \tau_2, 1) = (\sigma_2 z + 1, \tau_2 z, z) \\ \dot{T}_3 &= (0,1,0) + z \cdot (\sigma_3, \tau_3, 1) = (\sigma_3 z, \tau_3 z + 1, z) \end{aligned}$$

The affine combination $[\alpha, \beta]$ of the three generator rays that passes through P, is:

$$\dot{P} = (1 - \alpha - \beta) \cdot \dot{T}_1 + \alpha \cdot \dot{T}_2 + \beta \cdot \dot{T}_3 \qquad (13)$$

$[\alpha, \beta]$ can be computed using the ratio of the signed areas formed by triangle $\Delta \dot{T}_1 \dot{P} \dot{T}_3$, $\Delta \dot{T}_1 \dot{T}_2 \dot{P}$ over $\Delta \dot{T}_1 \dot{T}_2 \dot{T}_3$, as is shown in Figure 4. Notice the area formed by $\Delta \dot{T}_1 \dot{T}_2 \dot{T}_3$ corresponds to the characteristic equation of the GLC. Thus, the affine coefficients (α, β) can be computed as:

$$u = \alpha = \frac{\Delta \dot{T}_1 \dot{P} \dot{T}_3}{\Delta \dot{T}_1 \dot{T}_2 \dot{T}_3} = \frac{\begin{vmatrix} z\sigma_1 & z\tau_1 & 1 \\ x & y & 1 \\ z\sigma_3 & 1 + z\tau_3 & 1 \end{vmatrix}}{Az^2 + Bz + C} \qquad (14)$$

$$v = \beta = \frac{\Delta \dot{T}_1 \dot{T}_2 \dot{P}}{\Delta \dot{T}_1 \dot{T}_2 \dot{T}_3} = \frac{\begin{vmatrix} z\sigma_1 & z\tau_1 & 1 \\ 1 + z\sigma_2 & z\tau_2 & 1 \\ x & y & 1 \end{vmatrix}}{Az^2 + Bz + C}$$

Equation (14) and Figure 4 gives a geometric interpretation to the Projection Equation.

4.2   Singularities

Notice Equation (14) may lead to no solution or multiple solutions when the denominator $Az^2 + Bz + C = 0$ (i.e., the characteristic equation is zero). This happens when P lies at the depth of a slit. Thus, using Table 3.1, we can conclude that these singularities can only happen in cross-slits, pushbroom, pencil, and pinhole cameras.

When the points lie precisely on the slits, duplicated images will occur, because multiple GLC rays will pass through these points. The ray passing through the point is determined by the solution to a 4x4 system of equations given in (9) and (10). When the point lies on the slit, the determinant of this matrix is zero, and, therefore, the four equations become linearly dependent. For pinhole cameras, when the point coincides with the center of projection, the 4 linear equations will degenerate to 2 linear equations as (10) and the projection of the point will cover the whole image. For pushbroom, cross-slits, and pencils, the 4 linear constraints will degenerate to three independent equations, and the projection of each point on the singularity covers a 1D subspace of rays, or in its image it will project to as a line. A similar case happens with EPI cameras.

Furthermore, not all 3D points project onto a given GLC. There are two possible un-projectable situations: 1) there is no ray in the camera that can pass through the point, or 2) the ray that passes through the point is parallel to 2PP, and hence cannot be represented. Points that cannot be projected can only happen when the denominator of equation (14) is zero and the numerator is non-zero. For cross-slit cameras, these points lie on the two planes $\Pi_{z=z_1}$ and

$\Pi_{z=z_2}$ that contain the slits but do not lie on these slits. This is representative of the first case. For pencil and push-broom cameras, these singularity points lie on plane $\Pi_{z=z_1}$ that contains the slit but do not lie on the slit, and it follows the second case. Pinhole cameras are a special case of pencil cameras. In theory, it can image all points in 3D space. However, for points that lie on the plane parallel to the parametrization plane and passing through the COP, the corresponding rays are parallel to 2PP and hence cannot be imaged by a pinhole GLC.

4.3   Projections of Lines

Now we consider the projections of lines onto various GLCs. If l is parallel to the uv plane, we can parameterize l as a point $[x_0, y_0, z_0]$ on the line and the direction $[d^x, d^y, 0]$ of the line. All rays passing through l satisfy

$$[u, v, 0] + \lambda_1[\sigma, \tau, 1] = [x_0, y_0, z_0] + \lambda_2[d^x, d^y, 0] \quad (15)$$

It has been shown in [4] that equation (15) is equivalent to the linear constraint

$$(u + z_0\sigma - x_0)d^y - (v + z_0\tau - y_0)d^x = 0 \qquad (16)$$

The GLC rays passing through l is the intersection of three linear constraints: equation (9) and (16). Thus, the rays collected by any GLC passing through l are, in general, a 1D linear manifold. If we assume the uv plane is the default image plane, then (u, v) gives the pixel coordinates of the projection. This implies that the image of a line l parallel to the uv plane also a line.

If l is not parallel to the uv plane, then l will intersect uv plane at $(u_0, v_0, 0)$ and has direction $(\sigma_0, \tau_0, 1)$. All rays passing through l in this case satisfy the bilinear constraint [4]:

$$(u - u_0)(\tau - \tau_0) - (v - v_0)(\sigma - \sigma_0) = 0 \qquad (17)$$

The projection of l hence can be computed using equation (9) and equation (17) as follows:

$$(u - u_0)((1 - u - v)\tau_1 + u\tau_2 + v\tau_3 - \tau_0) \qquad (18)$$
$$-(v - v_0)((1 - u - v)\sigma_1 + u\sigma_2 + v\sigma_3 - \sigma_0) = 0$$

which corresponds to a 1D quadratic manifold of rays. Similarly, if we take the uv plane as the image plane, the image of l is a quadratic curve on the image plane as shown in Figure 2(a) and 2(b).

4.4   Projections of Points at Infinity

We can use the the properties of GLC line projection to determine the GLC projections of points lying in the plane at infinity. An infinite point can be written as:

$$P(x, y, z) = (u_0, v_0, 0) + z(\sigma_0, \tau_0, 1), z \to \infty \qquad (19)$$

Substituting P in equation (14), it is easy to see that the numerator and the denominator of u and v are both quadratic in z. However either or both terms may degenerate to linear or constant.

For pinhole, pencil, bilinear, and cross-slits, the first GLC characteristic equation (the denominator in the projection equation) is always general quadratic in z, as is shown in Table 1. And since the numerator is at most a quadratic in z, when $z \to \infty$, both u and v will have finite values, i.e., points infinitely far away from the image plane all have a projection in the camera.

Substituting $[x, y, z]$ in the Projection Equation (11), we get

$$u = \frac{A_u z^2 + B_u z + C_u}{z^2(s_1(t_2-t_3)+s_2(t_3-t_1)+s_3(t_1-t_2))-z(s_1-s_2+t_1-t_3)+1}$$
$$v = \frac{A_v z^2 + B_v z + C_v}{z^2(s_1(t_2-t_3)+s_2(t_3-t_1)+s_3(t_1-t_2))-z(s_1-s_2+t_1-t_3)+1}$$

where

$$A_u = \begin{vmatrix} \sigma_1 & \tau_1 & 1 \\ \sigma_0 & \tau_0 & 1 \\ \sigma_3 & \tau_3 & 1 \end{vmatrix}, A_v = \begin{vmatrix} \sigma_1 & \tau_1 & 1 \\ \sigma_2 & \tau_2 & 1 \\ \sigma_0 & \tau_0 & 1 \end{vmatrix} \quad (20)$$

Thus, the coefficients of z^2 in both the numerator and denominator of the projection equation are functions of σ_0, τ_0, and the intrinsic parameters of the GLC, not u_0 or v_0. This implies the final projection is only dependent on the direction of the infinite points. In the pinhole case, these points correspond to the vanishing points associated with directions and such vanishing points also exist for all pencil, bilinear, and cross-slits cameras.

For pushbroom cameras, the directions of three generator rays are parallel to some plane $\Pi_{pushbroom}$ and its characteristic equation is linear in z. The denominator in the Projection Equation (11) is, thus, a linear function in z. However, the numerator can be quadratic in z as shown in equation (20). Therefore, only when

$$A_u = \begin{vmatrix} \sigma_1 & \tau_1 & 1 \\ \sigma_0 & \tau_0 & 1 \\ \sigma_3 & \tau_3 & 1 \end{vmatrix} = 0, \quad A_v = \begin{vmatrix} \sigma_1 & \tau_1 & 1 \\ \sigma_2 & \tau_2 & 1 \\ \sigma_0 & \tau_0 & 1 \end{vmatrix} = 0 \quad (21)$$

can the point be projected into the camera. However, since the three generator rays are parallel to some plane, we must also have

$$\begin{vmatrix} \sigma_1 & \tau_1 & 1 \\ \sigma_2 & \tau_2 & 1 \\ \sigma_3 & \tau_3 & 1 \end{vmatrix} = 0 \quad (22)$$

From equation (21) and (22), $[\sigma_0, \tau_0, 1]$ must be a direction parallel to $\Pi_{pushbroom}$. Thus, the projection of the infinite points are constrained to one dimensional subspace and causes infinite stretching at the other, as is commonly observed in pushbroom panoramas. Cross-slit GLCs, however, are able to project all points infinitely far away and,
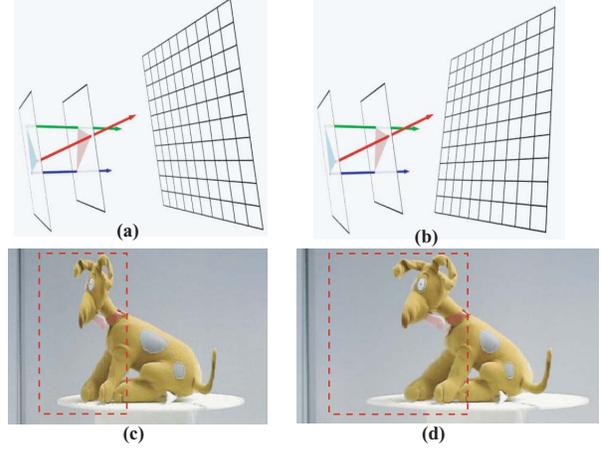
Figure 5. (a) Imaging the GLC on a plane parallel to the uv plane. (b) Imaging the GL on a different plane. (c) A GLC image of a toy dog synthesized from a video using image plane (a). (d) A different image of the same GLC camera using image plane (b).

therefore, are a better choice for creating panoramas. Figure 2(a) and 2(c) compares pushbroom and cross-slit panoramas. Objects far away are stretched in pushbroom cameras, but not in cross-slit cameras.

Similarly, for orthographic and twisted orthographic cameras, whose characteristic equations are constant, an infinite point has a projection only if $[\sigma_0, \tau_0, 1]$ is the direction of the ray of the GLC at point $[u_0, v_0, 0]$. For instance, for orthographic cameras, only infinite points along the view direction can be seen in the projection.

5 GLC Collineation

In the previous section, we study the projections of points and lines in a GLC by taking the uv plane as its default image plane. In this section, I derive how a given GLC image transforms as it is sampled over different image planes. This transformation is analogous to planar collineation (homography) to pinhole cameras.

Assume the new image plane Π_{new} is specified by an origin $\dot{p}$ and two spanning directions $\vec{d_1}$, $\vec{d_2}$. For every ray $r[\sigma, \tau, u, v]$ parameterized under the default 2PP, we intersect the ray r with Π_{new} to obtain its new pixel coordinate $[i_1, j_1]$:

$$[u, v, 0] + \lambda[\sigma, \tau, 1] = \dot{p} + \vec{d_1}i_1 + \vec{d_2}j_1 \quad (23)$$

To simplify our computation, we translate origin $\dot{p}$ on the plane to simply our computation. If we assume the new image plane is not parallel to the z axis, we can choose $\dot{p} = [0, 0, p^z]$. Otherwise, we can choose the image plane's intersections with either the x or y axis, and similar results hold. The new image coordinate of the ray $[i_1, j_1]$ can be

computed by solving equation (23) as:

$$i_1 = \frac{p^z(d_2^y\sigma - d_2^x\tau) + d_2^y u - d_2^x v + d_2^z(\sigma v - \tau u)}{\gamma} \quad (24)$$

$$j_1 = \frac{p^z(d_1^x\tau - d_1^y\sigma) + d_1^x v - d_1^y u - d_1^z(\sigma v - \tau u)}{\gamma}$$

where

$$\gamma = \begin{vmatrix} d_1^x & d_1^y & d_1^z \\ d_2^x & d_2^y & d_2^z \\ \sigma & \tau & 1 \end{vmatrix} \quad (25)$$

Since all rays lie in a GLC satisfy equation (9), thus, σ and τ are linear functions in u and v. Furthermore, γ is linear in σ and τ, and, therefore, can also be written as $\gamma = a_3 u + b_3 v + c_3$. Thus, i_1 and j_1 can be computed as:

$$i_1 = \frac{a_1 u^2 + b_1 uv + c_1 v^2 + d_1 u + e_1 v + f_1}{a_3 u + b_3 v + c_3} \quad (26)$$

$$j_1 = \frac{a_2 u^2 + b_2 uv + c_2 v^2 + d_2 u + e_2 v + f_2}{a_3 u + b_3 v + c_3}$$

γ is zero when the ray is parallel to the new parametrization plane, which we will not consider. The numerator is, in general, a quadratic function in u^2, v^2, uv, and, therefore, the collineation from the default image plane (uv plane) to a new image plane is quadratic rational.

5.1 General Collineation between GLCs

Finally, we can compute the collineation between any two planar parameterizations of the same GLC. Suppose we want to compute the collineation from image plane $\Pi_1 = \{\dot{p}, \vec{d}_1, \vec{d}_2\}$ with pixel coordinate $[i_1, j_1]$ to image plane $\Pi_2 = \{\dot{q}, \vec{d}_3, \vec{d}_4\}$ with pixel coordinate $[i_2, j_2]$. First, we compute the transformation from Π_1 to the default image plane. Second we use the transformation from the default image plane to Π_2.

The first transformation is in fact simply projecting each point on Π_1 onto the default plane, which can be described by the Projection Equation (11). Consider each pixel $m[i_1, j_1]$ corresponds to point M

$$\dot{M}[x, y, z] = \dot{p} + \vec{d}_1 i_1 + \vec{d}_2 j_1 = \begin{bmatrix} p_1^x + i_1 d_1^x + j_1 d_2^x \\ p_1^y + i_1 d_1^y + j_1 d_2^y \\ p_1^z + i_1 d_1^z + j_1 d_2^z \end{bmatrix} \quad (27)$$

The projection of M onto the uv plane is quadratic rational in $[x, y, z]$ and $[x, y, z]$ is linear in $[i_1, j_1]$ as shown in equation (11), therefore, the composed transformation can written as

$$u = \frac{f_1(i_1, j_1)}{\theta(i_1, j_1)} \quad v = \frac{f_2(i_1, j_1)}{\theta(i_1, j_1)} \quad (28)$$

where

$$f_1(i_1, j_1) = A_1 i_1^2 + B_1 i_1 j_1 + C_1 j_1^2 + D_1 i_1 + E_1 j_1 + F_1$$
$$f_2(i_1, j_1) = A_2 i_1^2 + B_2 i_1 j_1 + C_2 j_1^2 + D_2 i_1 + E_2 j_1 + F_2$$
$$\theta(i_1, j_1) = A_3 i_1^2 + B_3 i_1 j_1 + C_3 j_1^2 + D_3 i_1 + E_3 j_1 + F_3$$

Thus, the transformation from $[u, v]$ is quadratic rational in i_1 and j_1.

Next, we compute the Collineation from $[u, v]$ to $\Pi_2[i_2, j_2]$ using equation (26) and result is a composition of two quadratic rational functions, which is a quartic rational function that has the form:

$$i_2 = \frac{H_1(i_1, j_1)}{I(i_1, j_1)}, \quad j_2 = \frac{H_2(i_1, j_1)}{I(i_1, j_1)} \quad (29)$$

where $H_1(i_1, j_1)$, $H_2(i_1, j_1)$ and $I(i_1, j_1)$ are all quartic (4th order) polynomials in i_1 and j_1.

In Figure 5, we show two images (in red rectangle) of the same GLC camera using two image planes. The GLC used is a cross-slit. This indicates, although the collineation is quartic rational, the behavior of changing the image plane is similar to homography in a pinhole image.

5.2 Collineation Degeneracies

The planar collineation of a GLC is, in general, a quartic rational function. However, it may degenerate into a quadratic or even a linear rational function for specific types of GLCs. In this section, the form of special collineations for cross-slit, pushbroom, and pinhole cameras are considered.

First, consider equation (24), which maps rays parameterized onto an image plane Π_1 as pixel $[i_1, j_1]$, as a function of u, v, σ, and τ. Given a second image plane Π_2, we can represent u, v, σ, and τ in terms of its pixel coordinate $[i_2, j_2]$, and directly compute the collineation from Π_2 to Π_1.

If the Π_2 is parallel to the default 2PP, then u, v, σ, and τ are linear functions of i_2 and j_2 [17]. Substituting them into equation (24) gives a quadratic rational (quadratic in the numerator and linear in the denominator) collineation. If Π_2 is not parallel to the default 2PP, then it is non-trivial to represent σ, τ, u, and v in terms of i_2 and j_2 and σ, τ, u, and v are no longer linear functions of i_2 and j_2. However, for some special types of GLCs, such as pinhole, pushbroom, and cross-slits, it is still possible to represent σ, τ, u, and v in a simpler form of i_2 and j_2.

For cross-slit cameras, if Π_2 is not parallel to the default 2PP, then the two slits will intersect the new 2PP and can be represented as two lines $l_1[\sigma_1, \tau_1, u_1, v_1]$ and $l_2[\sigma_2, \tau_2, u_2, v_2]$. If we choose Π_2 as the parametrization plane, then each ray can be represented as $[\sigma, \tau, i_2, j_2]$. Recall that all rays passing through l_1 and l_2 are subject to two bilinear constraints (17). Thus:

$$(i_2 - u_1)(\tau - \tau_1) - (j_2 - v_1)(\sigma - \sigma_1) = 0 \quad (30)$$
$$(i_2 - u_2)(\tau - \tau_2) - (j_2 - v_2)(\sigma - \sigma_2) = 0$$

we can solve σ and τ in terms of i_2 and j_2 as:

$$\sigma = \frac{A_2 i_2^2 + B_2 i_2 j_2 + C_2 j_2^2 + D_2 i_2 + E_2 j_2 + F_2}{A_1 i_2 + B_1 j_2 + C_1} \quad (31)$$

$$\tau = \frac{A_3 i_2^2 + B_3 i_2 j_2 + C_3 j_2^2 + D_3 i_2 + E_3 j_2 + F_3}{A_1 i_2 + B_1 j_2 + C_1}$$

where $A_i, B_i, C_i, D_i, E_i, F_i, i = 1, 2, 3$ are all constant.

Expanding equation (30) gives

$$i_2\tau - j_2\sigma = A_4 i_2 + B_4 j_2 + C_4 \sigma + D_4 \tau + E_4 \qquad (32)$$

Finally, if we substitute σ, τ, and $i_2\tau - j_2\sigma$ into equation (24), the quartic rational collineation degenerates to quadratic rational (quadratic in numerator and denominator). An alternative proof using a quadratic tensor is shown in [16].

For pushbroom cameras, if Π_2 is not parallel to the default 2PP, then the slit will intersect the new 2PP as the line $[\sigma_1, \tau_1, u_1, v_2]$, and the new ray parametrization $[\sigma, \tau, i_2, j_2]$ is subject to one bilinear constraint, as is the case for the cross-slit GLCs:

$$(i_2 - u_1)(\tau - \tau_1) - (j_2 - v_1)(\sigma - \sigma_1) = 0 \qquad (33)$$

In addition, since all rays of a pushbroom GLC are parallel to some plane, all rays must also satisfy a second constraint

$$[\sigma, \tau, 1] \cdot [n_0, n_1, n_2] = n_0\sigma + n_1\tau + n_2 \qquad (34)$$

Solving for σ and τ of equation (33) and (34) gives

$$\sigma = \frac{A_2 i_2 + B_2 j_2 + C_2}{A_1 i_2 + B_1 j_2 + C_1} \qquad (35)$$
$$\tau = \frac{A_3 i_2 + B_3 j_2 + C_3}{A_1 i_2 + B_1 j_2 + C_1}$$

Similarly, we can substitute σ, τ, and $i_2\tau - j_2\sigma$ in equation (24), and the quartic rational collineation degenerates to a quadratic rational (quadratic in numerator and linear in denominator).

For pinhole cameras, rays in the GLC satisfy equation (10) as $i_2 + p^z\sigma = p^x$ and $j_2 + p^z\tau = p^y$. We can solve σ and τ as $\sigma = -\frac{1}{p^z} i_2 + \frac{p^x}{p^z}$ and $\tau = -\frac{1}{p^z} j_2 + \frac{p^y}{p^z}$. If we substituting σ and τ into the collineation equation, the quartic rational collineation degenerates to a linear rational (a homography).

Finally, for bilinear, pencil, twisted-orthographic GLCs, when the first image plane is not parallel to the default 2PP, it is non-trivial to solve σ and τ in term of i_2 and j_2 in general. And the quartic rational collineation will not reduce into simpler form.

6. Conclusions

We have presented theories of projection and collineation for GLC images. A closed-form solution for projecting 3D points to rays in a GLC was derived. In addition, we showed that colllneations between the images of the same GLC but on different imaging planes are up to quartic (4th order) rational. Using GLC projection and collineation theory, it is easy to explain the multiperspective distortions such as curving of lines, apparent stretching and shrinking, and duplicated projections of a single point.

References

[1] Simon Baker and Shree K. Nayar, "A Theory of Single-Viewpoint Catadioptric Image Formation, " *International Journal on Computer Vision*, 1999.

[2] P.Rademacher and G.Bishop, "Multiple-center-of-Projection Images," *Proc. ACM SIGGRAPH '98 (1998)* 199–206.

[3] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen, "The Lumigraph," *Proc. ACM SIGGRAPH '96 (1996)*, pp. 43–54.

[4] Xianfeng Gu, Steven J. Gortler, and Michael F. Cohen. Polyhedral geometry and the two-plane parameterization. Eurographics Rendering Workshop 1997 (1997) pages 1-12.

[5] R.I. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision," Cambridge Univ. Press, 2000.

[6] M. Levoy and P. Hanrahan,"Light Field Rendering," *Proc. ACM SIGGRAPH '96 (1996)*, 31–42.

[7] T. Pajdla, "Stereo with Oblique Cameras," *Int'l J. Computer Vision*, vol. 47, nos. 1/2/3 (2002) 161–170.

[8] T. Pajdla, "Geometry of Two-Slit Camera," *Research Report CTU–CMP–2002–02*, March 2002.

[9] S. Peleg, M. Ben-Ezra, and Y. Pritch, "Omnistereo: Panoramic Stereo Imaging," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 3 (2001) 279–290.

[10] R. Gupta and R.I. Hartley, "Linear Pushbroom Cameras," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 9 (1997) 963–975.

[11] S.M. Seitz, "The Space of All Stereo Images," *Proc. Int'l Conf. Computer Vision '01*, vol. I (2001) 26–33.

[12] J. Semple and G. Kneebone, "Algebraic Projective Geometry," Oxford: Clarendon Press, 1998.

[13] Rahul Swaminathan, Michael D. Grossberg and Shree K. Nayar, "Caustics of Catadioptric Cameras," *Proc. of IEEE International Conference on Computer Vision*, Vancouver, Canada, July 2001.

[14] Rahul Swaminathan, Michael D. Grossberg and Shree K. Nayar, "A Perspective on Distortions," *In Proc. IEEE Computer Vision and Pattern Recognition*, Wisconsin, June 2003.

[15] D. Wood, A. Finkelstein, J. Hughes, C. Thayer, and D. Salesin," Multiperspective Panoramas for Cel Animation," *Proc. ACM SIGGRAPH '97 (1997)* 243-250.

[16] A. Zomet, D. Feldman, S. Peleg, and D. Weinshall, "Mosaicing New Views: The Crossed-Slits Projection," *IEEE Trans. on PAMI (2003)* 741–754.

[17] J. Yu and L. McMillan, "General Linear Cameras," *Proc. 8th European. Conf. on Computer Vision (ECCV 2004)*, Volume II 14-27.

[18] J. Yu and L. McMillan, "Modelling Reflections via Multiperspective Imaging," *In Proc. IEEE Computer Vision and Pattern Recognition*, San Diego, June 2005.

[19] D. Zorin and A. H. Barr, "Correction of Geometric Perceptual Distortions in Pictures," *Computer Graphics*, 29:257–264, 1995.