

MULTIPERSPECTIVE MODELING AND RENDERING USING GENERAL LINEAR CAMERAS

JINGYI YU*, YUANYUAN DING*, AND LEONARD MCMILLAN†

Abstract. We present a General Linear Camera (GLC) model that unifies many previous camera models into a single representation. The GLC model is capable of describing all perspective (pinhole), orthographic, and many multiperspective (including pushbroom and two-slit) cameras, as well as epipolar plane images. It also includes three new and previously unexplored multiperspective linear cameras. The GLC model is both general and linear in the sense that, given any vector space where rays are represented as points, it describes all 2D affine subspaces (planes) that can be formed by affine combinations of 3 rays. The incident radiance seen along the rays found on subregions of these 2D linear subspaces are a precise definition of a projected image of a 3D scene. We model the GLC imaging process in terms of two separate stages: the mapping of 3D geometry to rays and the sampling of these rays over an image plane. We derive a closed-form solution to projecting 3D points in a scene to rays in a GLC and a notion of GLC collineation analogous to pinhole cameras. Finally, we develop a GLC ray-tracer for the direct rendering of GLC images. The GLC ray-tracer is able to create a broad class of multiperspective effects and it provides flexible collineation controls to reduce multiperspective distortions.

1. Introduction. Camera models are fundamental to the fields of computer vision and photogrammetry. The classic pinhole and orthographic camera models have long served as the workhorse of 3D imaging applications. However, recent developments have suggested alternative multiperspective camera models [6, 23] that provide alternate and potentially advantageous imaging systems for understanding the structure of observed scenes. Images captured by these cameras can effectively depict, within a single context, details of a scene that are simultaneously inaccessible from a single view, yet easily interpretable by a viewer. Researchers have further shown that these multiperspective cameras are amenable to stereo analysis and interpretation [16, 13, 23].

In contrast to pinhole and orthographic cameras, which can be completely characterized using a simple linear model (the classic 3 by 4 matrix [7]), multiperspective cameras models are defined less precisely. In practice, multiperspective cameras models are often described by constructions. By this we mean that a system or process is described for generating each specific class. While such physical models are useful for both acquisition and imparting intuition, they are not particularly amenable to analysis.

*Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716, USA, E-mail: {yu,ding}@eecis.udel.edu

†Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC 27911, USA. E-mail: mcmillan@cs.unc.edu

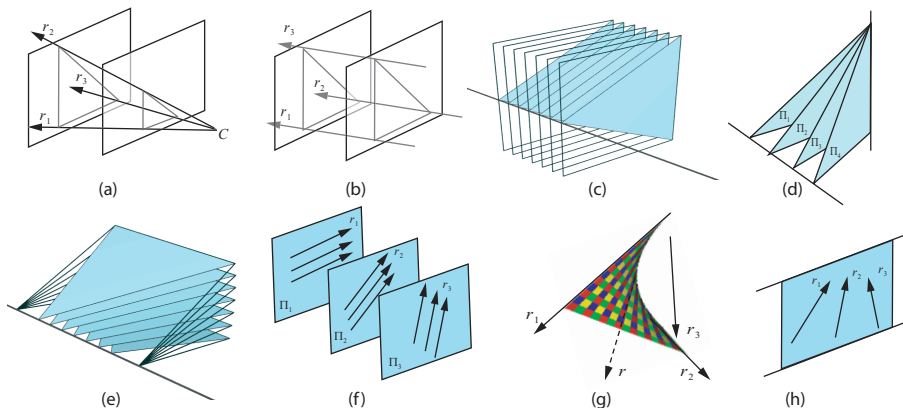


FIG. 1. *General Linear Camera Models.* (a) In a pinhole camera, all rays pass through a single point. (b) In an orthographic camera, all rays are parallel. (c) In a pushbroom, all rays lie on a set of parallel planes and pass through a line. (d) In a cross slit camera, all rays pass through two non-coplanar lines. (e) In a pencil camera, all coplanar rays originate from a point on a line and lie on a specific plane through the line. (f) In a twisted orthographic camera, all rays lie on parallel twisted planes and no rays intersect. (g) In a bilinear camera, no two rays are coplanar and no two rays intersect. (h) In an EPI camera, all rays lie on a 2D plane.

In this paper, we present a General Linear Camera (GLC) model that unifies many previous camera models into a single representation. The GLC model is capable of describing all perspective (pinhole), orthographic, and many multiperspective (including pushbroom and two-slit) cameras, as well as epipolar plane images. The GLC model is both general and linear in the sense that, given any vector space where rays are represented as points, it describes all 2D linear subspaces (planes) formed by the affine combination of 3 rays. The incident radiance seen along the rays of these 2D affine subspaces are a precise definition of a projected image of a 3D scene.

We model the GLC imaging process in terms of 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 and a notion of GLC collineation analogous to pinhole cameras to allow flexible sampling of the captured rays over an image plane. Finally, we develop a GLC ray-tracer for the direct rendering of GLC images. The GLC ray-tracer is able to create a broad class of multiperspective effects and it provides flexible collineation controls to reduce multiperspective distortions.

2. Previous Work. The most common linear camera model is the classic 3 x 4 pinhole camera matrix [7], which combines six extrinsic and five intrinsic camera parameters into 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

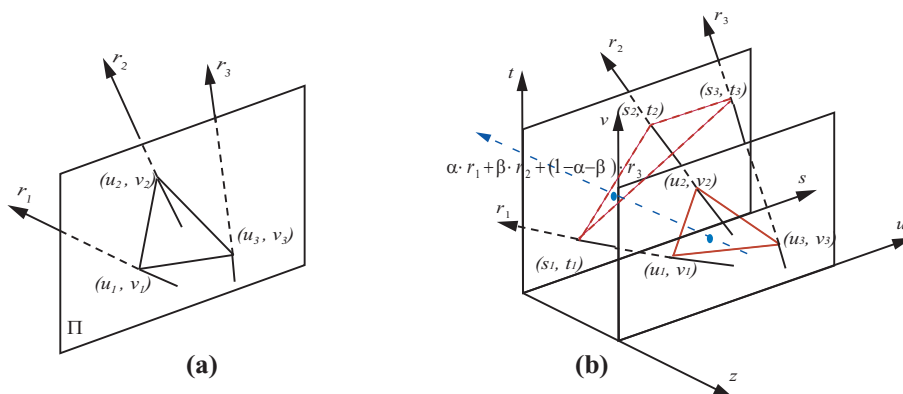


FIG. 2. *General Linear Camera Model.* a) A GLC is characterized by three rays originated from the image plane. b) It collects all possible affine combination of three rays.

can also be used to describe orthographic cameras. Recently, several researchers have proposed alternative camera representations known as multiperspective cameras which capture rays from different points in space. These multiperspective cameras include pushbroom cameras [6], which collect rays along parallel planes from points swept along a linear trajectory (Fig.1(c)), and cross-slit cameras [12], which collect all rays passing through two lines (Fig.1(d)). Zomet et al [23] did an extensive analysis and modelling of two slit(XSlit) multiperspective cameras. However, they discuss the relationship of these cameras to pinhole cameras only for the purpose of image construction, whereas we provide a unifying model.

Multiperspective camera models have also been explored in the field of computer graphics. Examples include multiple-center-of-projection images [14], manifold mosaics [13], and multiperspective panoramas [21, 1, 15]. Most multiperspective images are generated by stitching together parts of pinhole images [21, 14], or slicing through image sequences [13, 23]. Multiperspective rendering techniques are frequently employed by artists to depict more than can be seen from any specific point. Classic examples include the visual paradoxes of Escher, and the Cubism of Picasso and Braque. Multiperspective images have also been used as backdrops in cel animation to effectively depict camera motion in a single panorama [21].

Seitz [16] has analyzed the space of multiperspective cameras to determine those with a consistent epipolar geometry. Their work suggests that some multiperspective images can be used to analyze three-dimensional structure, just as pinhole cameras are commonly used. We focus our attention on a specific class of linear multiperspective cameras, most of which can be used to synthesize stereo or near stereo pairs [3].

Our analysis is closely related to the work of Gu et al [5], which explored the linear structures of 3D rays under a particular 4D mapping known as a two-plane

parametrization. This model is commonly used for light field rendering. Their primary focus was on the duality of points and planes under this mapping. They deduced that XSlits are another planar structure within this space, but they do not characterize all of the possible planar structures, nor discuss their analogous camera models.

We focus on identifying *all possible* 2D subspaces of rays and their corresponding camera models. In particular, we show in Section 3 that, besides cross-slit, pushbroom, and pinhole cameras, there exist several important but unexplored subsets of multiperspective cameras that also correspond to 2D linear subspaces of rays. We show these cameras can be uniquely characterized using a set of *ray characteristic equations*. Furthermore, [5] only cares about the set of rays lie on a 2D subspace while our work also studies the projection and sampling of these rays on the image plane.

3. General Linear Camera Model. The General Linear Camera (GLC) is defined by three rays that originate from three points $p_1(u_1, v_1)$, $p_2(u_2, v_2)$ and $p_3(u_3, v_3)$ on an image plane Π_{image} , as is shown in Fig.2. A GLC collects radiance measurements along all possible “affine combinations” of these three rays. In order to define this affine combination of rays, we assume a specific ray parametrization.

W.o.l.g, we define Π_{image} to lie on $z = 0$ plane and its origin to coincide with the origin of the coordinate system. From now on, we call Π_{image} as Π_{uv} . In order to parameterize rays, we place a second plane Π_{st} at $z = 1$. All rays not parallel to Π_{st}, Π_{uv} will intersect the two planes at $(s, t, 1)$ and $(u, v, 0)$ respectively. That gives a 4D parametrization of each ray in form (s, t, u, v) . This parametrization for rays, called the two-plane parametrization (2PP), is widely used by the computer graphics community for representing light fields and lumigraphs [9, 4]. Under this parametrization, an affine combination of three rays $r_i(s_i, t_i, u_i, v_i)$, $i = 1, 2, 3$, is defined as:

$$(1) \quad r = \alpha \cdot (s_1, t_1, u_1, v_1) + \beta \cdot (s_2, t_2, u_2, v_2) + (1 - \alpha - \beta) \cdot (s_3, t_3, u_3, v_3)$$

The choice of Π_{st} at $z = 1$, is, of course, arbitrary. One can choose any plane parallel to Π_{uv} to derive an equivalent parametrization. Moreover, these alternate parameterizations will preserve affine combinations of three rays.

LEMMA 1. *The affine combinations of any three rays under two different 2PP parameterizations that differ by choice of Π_{st} (i.e., (s, t, u, v) and (s', t', u, v)) are the same.*

Proof. Suppose $\Pi_{s't'}$ is at some arbitrary depth z_0 , $z_0 \neq 0$. Consider the transformation of a ray between the default parametrization ($z_0 = 1$) and this new one. If $r(s, t, u, v)$ and $r(s', t', u, v)$ represent the same ray r in 3D, then $r(s, t, u, v)$ must

pass through (s', t', z_0) , and there must exist some λ such that

$$\lambda \cdot (s, t, 1) + (1 - \lambda) \cdot (u, v, 0) = (s', t', z_0)$$

Solving for λ , we have

$$(2) \quad s' = s \cdot z_0 + u \cdot (1 - z_0), \quad t' = t \cdot z_0 + v \cdot (1 - z_0)$$

Since this transformation is linear, and affine combinations are preserved under linear transformation, the affine combinations of rays under our default two-plane parametrization ($z_0 = 1$) will be consistent for parameterizations over alternative parallel planes. Moreover, the affine weights for a particular choice of parallel Π_{st} are general. \square

3.1. Linearity. We call the GLC model “linear” because it defines all 2-dimensional affine subspaces in the 4-dimensional “ray space” imposed by a two-plane parametrization. Moreover, these 2D affine subspaces of rays can be considered as images. We refer to the three rays used in a particular GLC as the GLC’s *generator rays*. Equivalently, a GLC can be described by the coordinates of two triangles with corresponding vertices, one located on Π_{st} , and the second on Π_{uv} . Unless otherwise specified, we will assume the three generator rays (in their 4D parametrization) are linearly independent. This affine combination of generator rays also preserves linearity, while other parameterizations, such as the 6D *Plücker* coordinates [19], do not [5].

LEMMA 2. *If three rays are parallel to a plane Π in 3D, then all affine combinations of them are parallel to Π as well.*

LEMMA 3. *If three rays intersect a line l parallel to the image plane, all affine combinations of them will intersect l as well.*

Proof. By lemma 1, we can reparameterize three rays by placing Π_{st} so that it contains l resulting in the same set of affine combinations of the three rays. Because the st plane intersections of the three rays will lie on l , all affine combinations of three rays will have their st coordinates on l , i.e., they will all pass through l . The same argument can be applied to all rays which pass through a given point. \square

3.2. Equivalence of Classic Camera Models. Traditional camera models have equivalent GLC representations.

Pinhole camera: By definition, all rays of a pinhole camera pass through a single point, C in 3D space (the center of projection). Any three linearly independent rays from C will intersect Π_{uv} and Π_{st} planes to form two triangles. These triangles will be similar and have parallel corresponding edges, as shown in Fig.1(a). Furthermore, any other ray, r , through C will intersect Π_{uv} and Π_{st} planes at points \hat{p}_{uv} , and \hat{q}_{st} . These points will have the same affine coordinates relative to the triangle vertices on their corresponding planes, and r has the same affine coordinates as these two points.

Orthographic camera: By definition, all rays on an orthographic camera have the same direction. Any three linearly independent rays from an orthographic camera intersect parallel planes at the vertices of congruent triangles with parallel corresponding edges, as shown in Fig.1(b). Rays connecting the same affine combination of these triangle vertices, have the same direction as the 3 generator rays, and will, therefore, originate from the same orthographic camera.

Pushbroom camera: A pushbroom camera sweeps parallel planes along a line l collecting those rays that pass through l . We refer to this family of parallel planes as Π^* . We choose Π_{uv} parallel to l but not containing l , and select a non-degenerate set of generator rays (they intersect Π_{uv} in a triangle). By Lemma 2 and 3, all affine combinations of the three rays must all lie on Π^* parallel planes and must also pass through l and, hence, must belong to the pushbroom camera. In the other direction, for any point \dot{p} on Π_{uv} , there exist one ray that passes through \dot{p} , intersects l and is parallel to Π^* . Since \dot{p} must be some affine combination of the three vertexes of the uv triangle, r must lie on the corresponding GLC. Furthermore, because all rays of the pushbroom camera will intersect Π_{uv} , the GLC must generate equivalent rays.

XSlit camera: By definition, an XSlit camera collects all rays that pass through two non-coplanar lines. We choose Π_{uv} to be parallel to both lines but to not contain either of them. One can then pick a non-degenerate set of generator rays and find their corresponding triangles on Π_{st} and Π_{uv} . By Lemma 3, all affine combinations of these three rays must pass through both lines and hence must belong to the XSlit camera. In the other direction, authors of XSlit [12, 23] have shown that each point \dot{p} on the image plane Π_{uv} , maps to a unique ray r in an XSlit camera. Since \dot{p} must be some affine combination of the three vertexes of the uv triangle, r must belong to the GLC. The GLC hence must generate equivalent rays as the XSlit camera.

Epipolar Plane Image: EPI [2] cameras collect all rays that lie on a plane in 3D space. We therefore can pick any three linearly independent rays on the plane as generator rays. Affine combinations of these rays generate all possible rays on the plane, so long as they are linearly independent. Therefore a GLC can also represent Epipolar Plane Images.

3.3. GLCs under Relative Two-Plane-Parametrization. GLCs are defined as the affine subspaces of three generator rays that are parameterized by their intersections with two parallel planes at $[s, t, 1]$ and $[u, v, 0]$. The two-plane-parametrization of the ray can be viewed as a two-endpoint-parametrization of a line. Alternatively, each ray can be parameterized with a origin and a direction. If we choose the origin as $[u, v, 0]$ and the direction as $[\sigma, \tau, 1] = [s, t, 1] - [u, v, 0]$, we have a point-direction-parametrization of ray as $[\sigma, \tau, u, v]$. Since $[\sigma, \tau, 1]$ can also be viewed as the relative coordinate of $[s, t, 1]$ with respect to $[u, v, 0]$, we refer to this parametrization as the

relative-two-plane-parametrization (R2PP).

Notice, the transformation from 2PP to R2PP is linear, and more precisely a *shearing* transformation. Thus, affine subspaces are preserved under R2PP, and we can reparameterize a GLC by representing the rays under R2PP as:

$$(3) \quad GLC = \alpha \cdot [\sigma_1, \tau_1, u_1, v_1] + \beta \cdot [\sigma_2, \tau_2, u_2, v_2] + (1 - \alpha - \beta) \cdot [\sigma_3, \tau_3, u_3, v_3]$$

For the remaining of this paper, we will use $[\sigma, \tau, u, v]$ to parameterize rays and to represent the GLCs.

3.4. Canonical GLC Representations. A GLC defined under Equation (3) (or (1)) takes 12 variables. However, this GLC representation is not unique since one can pick a different set of three rays to define the same GLC. To enforce uniqueness, we can choose three generator rays in a specific form as $[\sigma_1, \tau_1, 0, 0]$, $[\sigma_2, \tau_2, 1, 0]$, and $[\sigma_3, \tau_3, 0, 1]$, i.e., the three rays originate from $[0, 0, 0]$, $[1, 0, 0]$, and $[0, 1, 0]$ from the default image (uv) plane. We call this representation the *canonical* GLC. Except for those GLCs that have a slit lie on the uv plane, all other GLCs can be uniquely represented using this 6-parameter canonical representation.

Since the uv plane corresponds to the default image plane, using the canonical GLC representation is similar to specifying the texture coordinate of a triangle, where the texture coordinate here represents the direction $[\sigma, \tau]$ of the ray. We will use the canonical representation of the GLC to study the GLC projection model in Section 6.

4. Characteristic Equation of GLC. We have shown that the GLC model can describe classical all pinhole, orthographic, pushbroom, and cross-slit cameras, as well as the EPIs. In this section we develop a criterion to classify a GLC. One discriminating characteristic of affine ray combinations is whether or not all rays pass through a line in 3D space. This characteristic is fundamental to the definition of many multi-perspective cameras. We will use this criteria to define the characteristic equation of general linear cameras.

Recall that any 2D affine subspace in 4D can be defined as affine combinations of three points. Thus, GLC models can be associated with all possible planes in the 4D since GLCs are specified as affine combinations of three rays, whose duals in 4D are the three points.

LEMMA 4. *Given a non-EPI, non-pinhole GLC, if all camera rays pass through some line l , not at infinity, in 3D space, then l must be parallel to Π_{uv} .*

Proof. We demonstrate the contrapositive. If l is not parallel to Π_{uv} , and all rays on a GLC pass through l , then we show the GLC must be either an EPI or a pinhole camera.

Assume the three rays pass through at least two distinct points on l , otherwise, they will be on a pinhole camera, by Lemma 3. If l is not parallel, then it must intersect Π_{st} , Π_{uv} at some point $(s_0, t_0, 1)$ and $(u_0, v_0, 0)$. Gu et al [5] has shown all rays passing through l must satisfy the following bilinear constraints

$$(4) \quad (u - u_0)(t - t_0) - (v - v_0)(s - s_0) = 0$$

We show that the only GLCs that satisfy this constraint are EPIs or pinholes.

All 2D affine subspaces in (s, t, u, v) can be written as the intersection of two linear constraints $A_i \cdot s + B_i \cdot t + C_i \cdot u + D_i \cdot v + E_i = 0$, $i = 1, 2$. In general we can solve these two equations for two variables, for instance, we can solve for $u-v$ as

$$(5) \quad u = A'_1 \cdot s + B'_1 \cdot t + E'_1, \quad v = A'_2 \cdot s + B'_2 \cdot t + E'_2$$

Substituting u and v into the bilinear constraint (4), we have

$$(6) \quad (A'_1 \cdot s + B'_1 \cdot t + E'_1 - u_0)(t - t_0) = (A'_2 \cdot s + B'_2 \cdot t + E'_2 - v_0)(s - s_0)$$

This equation can only be satisfied for all s and t if $A'_1 = B'_2$ and $B'_1 = A'_2 = 0$, therefore, equation (5) can be rewritten as $u = A' \cdot s + E'_1$ and $v = A' \cdot t + E'_2$. Gu et al [5] have shown all rays in this form must pass through a 3D point P (P cannot be at infinity, otherwise all rays have uniform directions and cannot all pass through any line l , not at infinity). Therefore all rays must lie on a 3D plane that passes through l and finite P . The only GLC camera in which all rays lie on a 3D plane is an EPI. If the two linear constraints are singular in u and v , we can solve for $s-t$, and similar results hold.

If the two linear constraints cannot be solved for $u-v$ or $s-t$ but can be solved for $u-s$ or $v-t$, then a similar analysis results in equations of two parallel lines, one on Π_{st} , the other on Π_{uv} . The set of rays through two parallel lines must lie on an EPI. \square

Lemma 3 and 4 imply that given a GLC, we need only consider if the three generator rays pass through some line parallel to Π_{st} . We use this relationship to define the *characteristic equation* of a GLC.

Recall the three generator rays in a GLC intersect some plane $\Pi_{z=\lambda}$ parallel to Π_{uv} at

$$T_i = (u_i, v_i, 0) + \lambda \cdot (\sigma, \tau, 1) \quad i = 1, 2, 3$$

By Lemma 3, all rays on the GLC pass through some line l on $\Pi_{z=\lambda}$ if the three generator rays intersect l . Therefore, we only need to test if there exist any λ so that the three intersection points T_1 , T_2 , and T_3 lie on a line. A necessary and sufficient condition for 3 points on a constant z -plane to be co-linear is that they have form

area on that plane. This area is computed as follows:

$$(7) \quad \begin{vmatrix} u_1 + \lambda \cdot \sigma_1 & v_1 + \lambda \cdot \tau_1 & 1 \\ u_2 + \lambda \cdot \sigma_2 & v_2 + \lambda \cdot \tau_2 & 1 \\ u_3 + \lambda \cdot \sigma_3 & v_3 + \lambda \cdot \tau_3 & 1 \end{vmatrix} = 0$$

which results in a quadratic equation of the form

$$(8) \quad A\lambda^2 + B\lambda + C = 0$$

where

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

We call Equation (8) the characteristic equation of a GLC. Since the characteristic equation can be calculated from any three rays, one can also evaluate the characteristic equation for EPI and pinhole cameras. The number of solutions of the characteristic equation implies the number of lines that all rays on a GLC pass through. Since it is quadratic, it may have 0, 1, 2 or infinite solutions. The number of solutions depends on the denominator A and the quadratic discriminant $\Delta = B^2 - 4AC$.

4.1. Characterizing Classic Camera Models. We start with showing how to identify standard camera models using the characteristic equation of a GLC.

LEMMA 5. *Given a GLC, three generator rays, and its characteristic equation $A \cdot \lambda^2 + B \cdot \lambda + C = 0$, then all rays are parallel to some plane if and only if $A = 0$.*

Proof. Notice in the matrix used to calculate A , row i is the direction \vec{d}_i of ray r_i . Therefore A can be rewritten as $A = (\vec{d}_1 \times \vec{d}_2) \cdot \vec{d}_3$. Hence $A = 0$ if and only if \vec{d}_1 , \vec{d}_2 and \vec{d}_3 are parallel to some 3D plane. And by Lemma 2, all affine combinations of these rays must also be parallel to that plane if $A = 0$. □

4.2. $A = 0$ case. When $A = 0$, the characteristic equation degenerates to a linear equation, which can have 1, 0, or an infinite number of solutions. By Lemma 5, all rays are parallel to some plane. Only three standard camera models satisfy this condition: pushbroom, orthographic, and EPI.

All rays of a pushbroom lie on parallel planes and pass through one line, as is shown in Fig.1(c). A GLC is a pushbroom camera if and only if $A = 0$ and the characteristic equation has 1 solution.

All rays of an orthographic camera have the same direction and do not all simultaneously pass through any line l . Hence its characteristic equation has no solution. The zero solution criteria alone, however, is insufficient to determine if a GLC is orthographic. We show in the following section that one can *twist* an orthographic camera

into bilinear sheets by rotating rays on parallel planes, as is shown in Fig.1(f), and still maintain that all rays do not pass through a common line. In Section 3, we have shown that corresponding edges of the two congruent triangles of an orthographic GLC must be parallel. This parallelism is captured by the following expression:

$$(9) \quad \frac{(\sigma_i - \sigma_j)}{(\tau_i - \tau_j)} = \frac{(u_i - u_j)}{(v_i - v_j)} \quad i, j = 1, 2, 3 \quad \text{and} \quad i \neq j$$

We call this condition the *edge-parallel* condition. It is easy to verify that a GLC is orthographic if and only if $A = 0$, its characteristic equation has no solution, and it satisfies the *edge-parallel* condition.

Rays of an EPI camera all lie on a plane and pass through an infinite number of lines on the plane. In order for a characteristic equation to have infinite number of solutions when $A = 0$, we must also have $B = 0$ and $C = 0$. This is not surprising, because the intersection of the epipolar plane with Π_{st} and Π_{uv} must be two parallel lines and it is easy to verify $A = 0$, $B = 0$ and $C = 0$ if and only if the corresponding GLC is an EPI.

4.3. $A \neq 0$ case. When $A \neq 0$, the characteristic equation becomes quadratic and can have 0, 1, or 2 solutions, which depends on the characteristic equation's discriminant Δ . We show how to identify the remaining two classical cameras, pinhole and XSlit cameras in term of A and Δ .

All rays in a pinhole camera pass through the center of projection (COP). Therefore, any three rays from a pinhole camera, if linearly independent, cannot all be parallel to any plane, and by Lemma 4, $A \neq 0$. Notice that the roots of the characteristic equation correspond to *the depth* of the line that all rays pass through, hence the characteristic equation of a pinhole camera can only have one solution that corresponds to the depth of the COP, even though there exists an infinite number of lines passing through the COP. Therefore, the characteristic equation of a pinhole camera must satisfy $A \neq 0$ and $\Delta = 0$. However, this condition alone is insufficient to determine if a GLC is pinhole. In the following section, we show that there exists a camera where all rays lie on pencil of planes sharing a line, as shown in Fig.1(e), which also satisfies these conditions. One can, however, reuse the *edge-parallel* condition to verify if a GLC is pinhole. Thus a GLC is pinhole, if and only if $A \neq 0$, has one solution, and it satisfies *edge-parallel* condition.

Rays of an XSlit camera pass through two slits and, therefore, the characteristic equation of a GLC must have at least two distinct solutions. Furthermore, Pajdla [12] has shown all rays of an XSlit camera cannot pass through lines other than its two slits, therefore, the characteristic equation of an XSlit camera has exactly two distinct solutions. Thus, a GLC is an XSlit if and only if $A \neq 0$ and $\Delta > 0$.

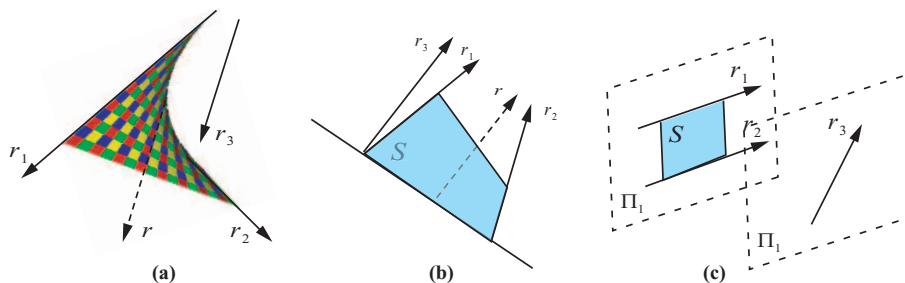


FIG. 3. *Bilinear Surfaces.* (a) r_3 is parallel to S ; (b) r_3 is parallel to S , but still intersects S ; (c) r_3 is not parallel to S , and does not intersect S either.

5. Characterizing New Camera Models. The characteristic equation also suggests three new multiperspective camera types that have not been previously discussed. They include 1)twisted orthographic: $A = 0$, the equation has no solution, and all rays do not have uniform direction; 2)pencil camera: $A \neq 0$ and the equation has one root, but all rays do not pass through a 3D point; 3)bilinear camera: $A \neq 0$ and the characteristic equation has no solution. In this section, we give a geometric interpretation of these three new camera models.

5.1. Ray Geometry. Before describing these camera models, however, we will first discuss a helpful interpretation of the spatial relationships between the three generator rays. An affine combination of two 4D points defines a 1-dimensional affine subspace. Under 2PP, a 1-D affine subspaces corresponds to a bilinear surface S in 3D that contains the two rays associated with each 4D point. If these two rays intersect or have the same direction in 3D space, S degenerates to a plane. Next, we consider the relationship between ray r_3 and S . We define r_3 to be parallel to S if and only if r_3 has the same direction as some ray $r \in S$. This definition of parallelism is quite different from conventional definitions. In particular, if r_3 is parallel to S , r_3 can still intersect S . And if r_3 is not parallel to S , r_3 still might not intersect S , Fig.3(b) and (c) show examples of each case.

This definition of parallelism, however, is closely related to A in the characteristic equation. If r_3 is parallel to S , by definition, the direction of r_3 must be some linear combination of the directions of r_1 and r_2 , and, therefore, $A = 0$ by Lemma 5. $A = 0$, however, is not sufficient to guarantee r_3 is parallel to S . For instance, one can pick two rays with uniform directions so that $A = 0$, yet still have the freedom to pick a third so that it is not parallel to the plane, as is shown in Fig.3(c).

The number of solutions to the characteristic equation is also closely related to the number of intersections of r_3 with S . If r_3 intersects the bilinear surface $S(r_1, r_2)$ at P , then there exists a line l , where $P \in l$, that all rays pass through. This is

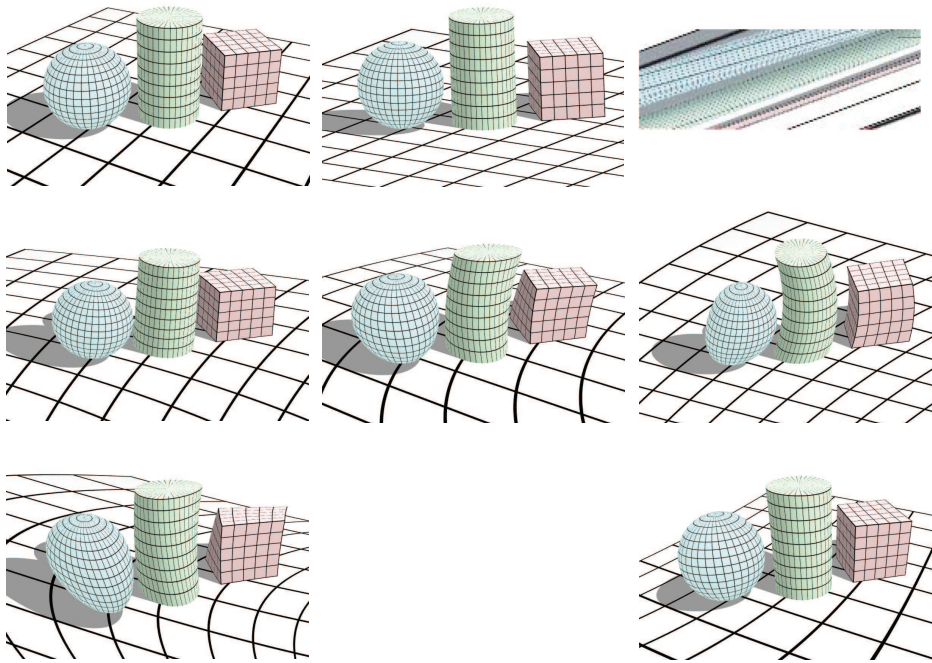


FIG. 4. Comparison between synthetic GLC images. From left to right, top row: a pinhole, an orthographic and an EPI; middle row: a pushbroom, a pencil and an twisted orthographic; bottom row: a bilinear and an cross-slit.

because one can place a constant- z plane that passes through P and intersects r_1 and r_2 at Q and R . It is easy to verify that P , Q and R lie on a line and, therefore, all rays must pass through line PQR . Hence r_3 intersecting $S(r_1, r_2)$ is a sufficient condition to ensure that all rays pass through some line. It further implies if the characteristic equation of a GLC has no solution, no two rays in the camera intersect. GLCs whose characteristic equation has no solution are examples of the oblique camera from [11].

5.2. New Multiperspective Cameras. Our GLC model and its characteristic equation suggests 3 new camera types that have not been previously described.

Twisted Orthographic Camera: The characteristic equation of the twisted orthographic camera satisfies $A = 0$, has no solution, and its generators do not satisfy the *edge-parallel* condition. If r_1 , r_2 and r_3 are linearly independent, no solution implies r_3 will not intersect the bilinear surface S . In fact, no two rays intersect in 3D space. In addition, $A = 0$ also implies that all rays are parallel to some plane Π in 3D space, therefore the rays on each of these parallel planes must have uniform directions as is shown in Fig.1(f). Therefore, twisted orthographic camera can be viewed as twisting parallel planes of rays in an orthographic camera along common bilinear sheets.

TABLE 1
Characterize General Linear Cameras by Characteristic Equation

Characteristic Equation	2 Solution	1 Solution	0 Solution	Inf. Solution
$A \neq 0$	XSlit	Pencil/Pinhole [†]	Bilinear	\emptyset
$A = 0$	\emptyset	Pushbroom	Twisted/Ortho. [†]	EPI

[†]: A GLC satisfying *edge-parallel* condition is pinhole ($A \neq 0$) or orthographic ($A = 0$).

Pencil Camera: The characteristic equation of a pencil camera satisfies $A \neq 0$, has one solution and the generators do not satisfy the *edge-parallel* condition. In Fig.1(e), we illustrate a sample pencil camera: rays lie on a pencil of planes that share line l . In a pushbroom camera, all rays also pass through a single line. However, pushbroom cameras collect rays along planes transverse to l whereas the planes of a pencil camera contains l (i.e., lie in the pencil of planes through l), as is shown in Fig.1(c) and 1(e).

Bilinear Camera: By definition, the characteristic equation of a bilinear camera satisfies $A \neq 0$ and the equation has no solution ($\Delta < 0$). Therefore, similar to twisted orthographic cameras, no two rays intersect in 3D in a bilinear camera. In addition, since $A \neq 0$, no two rays are parallel either. Therefore, any two rays in a bilinear camera form a non-degenerate bilinear surface, as is shown in Fig.3(a). The complete classification of cameras is listed in Table 1.

In Fig.4, we show the GLC images that are rendered using our GLC Ray Tracer (Section 7.2). Different types of GLCs exhibit unique multiperspective distortions as shown in the curved isolines on the objects. We analyze the cause of these distortions in the next Section.

6. GLC Projection and Collineation. Next, we study the GLC imaging process. We first consider projecting a 3D point onto a GLC. To simplify the analysis, we use the 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:

$$(10) \quad r[\sigma, \tau, u, v] = (1 - \alpha - \beta) \cdot [\sigma_1, \tau_1, 0, 0] + \alpha \cdot [\sigma_2, \tau_2, 1, 0] + \beta \cdot [\sigma_3, \tau_3, 0, 1]$$

where $\sigma_i, \tau_i, i = 1, 2, 3$ are constant for a given GLC. It is easy to see that $\alpha = u$ and $\beta = v$ under this simplification. It is also worth noting that Equation (10) is also equivalent to the following two linear constraints:

$$(11) \quad \begin{aligned} \sigma &= (1 - u - v)\sigma_1 + u\sigma_2 + v\sigma_3 \\ \tau &= (1 - u - v)\tau_1 + u\tau_2 + v\tau_3 \end{aligned}$$

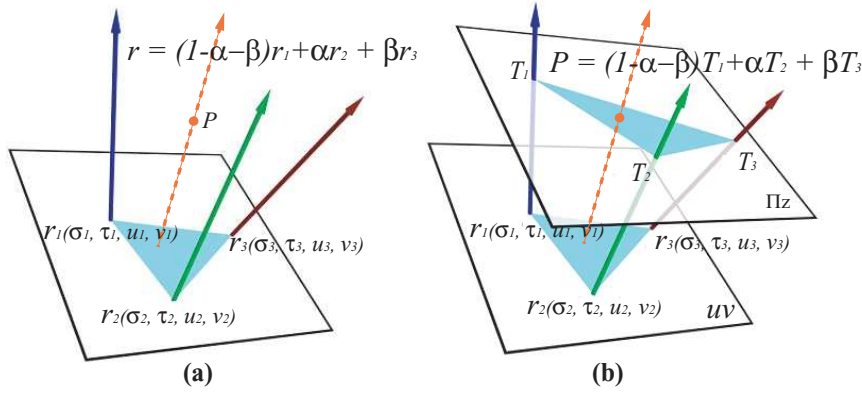


FIG. 5. (a) Projecting a point P to a ray in the GLC. (b) The projection of P can be computed using the same affine coordinate on the sweeping plane Π_z .

The GLC ray that passes through a 3D point $\dot{P}(x, y, z)$ satisfies the following linear constraints [5]:

$$(12) \quad \begin{aligned} u + z \cdot \sigma &= x \\ v + z \cdot \tau &= y \end{aligned}$$

The ray passing through P is, thus, the solution of the four equations in (11) and (12) and can be computed as:

$$(13) \quad \begin{aligned} u &= -\frac{(z^2(\sigma_1\tau_3 - \sigma_3\tau_1) - z(\sigma_1(y-1) - \sigma_3y - x(\tau_1 - \tau_3)) - x)}{Az^2 + Bz + C} \\ v &= \frac{(z^2(\sigma_1\tau_2 - \sigma_2\tau_1) - z(\sigma_1y - \sigma_2y + \tau_1(1-x) + \tau_2x) + y)}{Az^2 + Bz + C} \end{aligned}$$

where $Az^2 + Bz + C = 0$ corresponds to the characteristic of the GLC. We call Equation (13) the GLC Projection Equation.

6.1. Plane Sweeping. The GLC Projection Equation (13) also 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

$$(14) \quad \begin{aligned} \dot{T}_1 &= (0, 0, 0) + z \cdot (\sigma_1, \tau_1, 1) = (\sigma_1z, \tau_1z, z) \\ \dot{T}_2 &= (1, 0, 0) + z \cdot (\sigma_2, \tau_2, 1) = (\sigma_2z + 1, \tau_2z, z) \\ \dot{T}_3 &= (0, 1, 0) + z \cdot (\sigma_3, \tau_3, 1) = (\sigma_3z, \tau_3z + 1, z) \end{aligned}$$

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

$$(15) \quad \dot{P} = (1 - \alpha - \beta) \cdot \dot{T}_1 + \alpha \cdot \dot{T}_2 + \beta \cdot \dot{T}_3$$

$[\alpha, \beta]$ 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 Fig.5. 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:

$$(16) \quad \begin{aligned} u = \alpha &= \frac{\Delta\dot{T}_1\dot{P}\dot{T}_3}{\Delta\dot{T}_1\dot{T}_2\dot{T}_3} = \frac{\begin{vmatrix} z\sigma_1 & z\tau_1 & 1 \\ x & y & 1 \\ z\sigma_3 & 1+z\tau_3 & 1 \end{vmatrix}}{Az^2 + Bz + C} \\ v = \beta &= \frac{\Delta\dot{T}_1\dot{T}_2\dot{P}}{\Delta\dot{T}_1\dot{T}_2\dot{T}_3} = \frac{\begin{vmatrix} z\sigma_1 & z\tau_1 & 1 \\ 1+z\sigma_2 & z\tau_2 & 1 \\ x & y & 1 \end{vmatrix}}{Az^2 + Bz + C} \end{aligned}$$

Equation (16) and Fig.5(b) give a geometric interpretation to the GLC Projection Equation.

6.2. Singularities. Notice Equation (16) 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 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 (11) and (12). 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 (12) 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 (16) 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 pushbroom 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.

6.3. 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

$$(17) \quad [u, v, 0] + \lambda_1[\sigma, \tau, 1] = [x_0, y_0, z_0] + \lambda_2[d^x, d^y, 0]$$

It has been shown in [5] that equation (17) is equivalent to the linear constraint

$$(18) \quad (u + z_0\sigma - x_0)d^y - (v + z_0\tau - y_0)d^x = 0$$

The GLC rays passing through l is the intersection of three linear constraints: equation (11) and (18). 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 [5]:

$$(19) \quad (u - u_0)(\tau - \tau_0) - (v - v_0)(\sigma - \sigma_0) = 0$$

The projection of l hence can be computed using equation (11) and equation (19) as follows:

$$(20) \quad \begin{aligned} & (u - u_0)((1 - u - v)\tau_1 + u\tau_2 + v\tau_3 - \tau_0) \\ & - (v - v_0)((1 - u - v)\sigma_1 + u\sigma_2 + v\sigma_3 - \sigma_0) = 0 \end{aligned}$$

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 Fig.4.

6.4. 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:

$$(21) \quad P(x, y, z) = (u_0, v_0, 0) + z(\sigma_0, \tau_0, 1), z \rightarrow \infty$$

Substituting P in Equation (16), 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 $z \rightarrow \infty$, 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 GLC Projection Equation (13), we get

$$(22) \quad \begin{aligned} u &= \frac{A_u z^2 + B_u z + C_u}{z^2(s_1(t_2 - t_3) + s_2(t_3 - t_1) + s_3(t_1 - t_2)) - z(s_1 - s_2 + t_1 - t_3) + 1} \\ v &= \frac{A_v z^2 + B_v z + C_v}{z^2(s_1(t_2 - t_3) + s_2(t_3 - t_1) + s_3(t_1 - t_2)) - z(s_1 - s_2 + t_1 - t_3) + 1} \end{aligned}$$

where

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

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 (13) is, thus, a linear function in z . However, the numerator can be quadratic in z as shown in equation (22). Therefore, only when

$$(23) \quad A_u = \begin{vmatrix} \sigma_1 & \tau_1 & 1 \\ \sigma_0 & \tau_0 & 1 \\ \sigma_3 & \tau_3 & 1 \end{vmatrix} = 0, \quad A_v = \begin{vmatrix} \sigma_1 & \tau_1 & 1 \\ \sigma_2 & \tau_2 & 1 \\ \sigma_0 & \tau_0 & 1 \end{vmatrix} = 0$$

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

$$(24) \quad \begin{vmatrix} \sigma_1 & \tau_1 & 1 \\ \sigma_2 & \tau_2 & 1 \\ \sigma_3 & \tau_3 & 1 \end{vmatrix} = 0$$

From equation (23) and (24), $[\sigma_0, \tau_0, 1]$ 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, therefore, are a better choice for creating panoramas. Fig.6(a) and Fig.6(b) 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.

6.5. GLC Collineation. We have shown how to project 3D points and lines on to the default uv image plane in a GLC. Next, we derive how to resample the rays collected by a GLC over different image planes. This transformation is analogous to planar collineation (homography) to pinhole cameras.

A GLC collineation \tilde{Col}_Π maps every ray $r(u, v)$ to a pixel $[i, j]$ on the image plane $\Pi[\dot{p}, \vec{d}_1, \vec{d}_2]$, where \dot{p} specifies the origin and \vec{d}_1 , and \vec{d}_2 specify the two spanning directions of Π . For every ray $r[\sigma, \tau, u, v]$, we can intersect r with Π to compute $[i, j]$:

$$(25) \quad [u, v, 0] + \lambda[\sigma, \tau, 1] = \dot{p} + i\vec{d}_1 + j\vec{d}_2$$

Solving for i , j , and λ gives:

$$(26) \quad i = \frac{(\tau d_2^z - d_2^y)(u - p_x) + (d_2^x - \sigma d_2^z)(v - p_y) - (\sigma d_2^y - \tau d_2^x)p_z}{\gamma}$$

$$j = \frac{(d_1^y - \tau d_1^z)(u - p_x) + (\sigma d_1^z - d_1^x)(v - p_y) - (\tau d_1^x - \sigma d_1^y)p_z}{\gamma}$$

where

$$(27) \quad \gamma = \begin{vmatrix} d_1^x & d_2^x & -\sigma \\ d_1^y & d_2^y & -\tau \\ d_1^z & d_2^z & -1 \end{vmatrix}$$

For a canonical GLC, since σ and τ are both linear functions in u and v as shown in Equation (11), γ must be linear in u and v . Therefore, we can rewrite i and j as:

$$(28) \quad i = \frac{a_1 u^2 + b_1 uv + c_1 v^2 + d_1 u + e_1 v + f_1}{a_3 u + b_3 v + c_3}$$

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

Thus, the collineation \tilde{Col}_Π of a GLC from the uv image plane to a new image plane Π is a quadratic rational function. Fig. 6 shows the images of a GLC under different collineations. It implies that image distortions may be reduced using a proper collineation.

7. Rendering GLC Images. GLC images can be rendered directly by cutting through pre-captured light fields, or by ray tracing a synthetic scene.

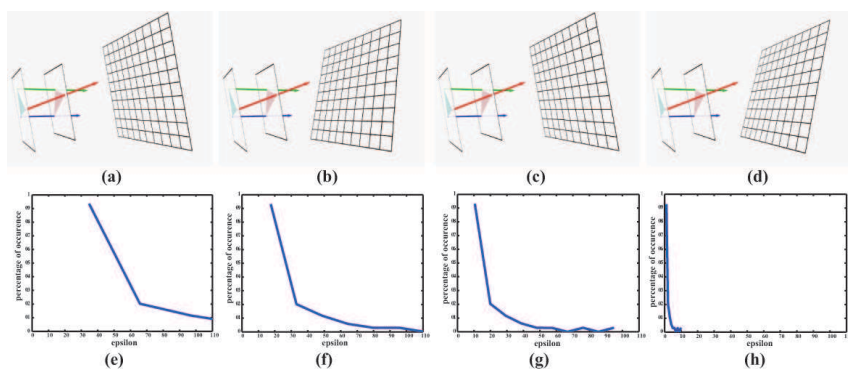


FIG. 6. *The image of a cross-slit GLC (d) under collineation (c) appear much less distorted than the image (b) of the same camera under collineation (a).*

7.1. Rendering GLCs From Light Fields. In Fig.8, we synthesize a cross-slit and a pencil camera by cutting through a densely sampled 3D light field along different directions. By appropriately organizing rays, both GLCs generate interpretable images similar to pinhole and orthographic cameras. The pencil camera synthesized in Fig.8 are twisted. This is because we are moving the viewing angle from left to right while scanning the image from the top to the bottom.

In Fig.7, we illustrate GLC images from a 4D light field. Each GLC is specified by three generator rays shown in red. By appropriately transforming the rays using a collineation, most GLCs generate easily interpretable images. Furthermore, we can use the light fields to navigate through the scene and to choose specific rays to form a desirable GLC. In Fig.9, we choose three rays from different perspectives and fuse them into a multiperspective bilinear GLC image.

In our implementation, we have used the common quadrilinear interpolations to interpolate the GLC rays from the light field. Since collecting all rays present in a scene is impractical or impossible for most light fields, aliasing artifacts called "ghosting" may appear in the rendered GLC images where the light field is undersampled, as shown on the background specularly in Fig.7.

7.2. General Linear Ray Tracer. We have also implemented a GLC Ray-Tracer (GLC-RT) based on the legacy Pov-Ray [24] framework. Our GLC-RT supports rendering both canonical GLC models and standard GLC models with arbitrary collineations.

The canonical GLC model has format:

```

GLC Camera{
    Generator Rays: <  $\sigma_1, \tau_1$  >, <  $\sigma_2, \tau_2$  >, <  $\sigma_3, \tau_3$  >
}

```



(a) Pushbroom



(b) Cross-Slit

FIG. 6. *GLC images synthesized from a 3D light field: (a) is a pushbroom panorama, (b) is a cross-slit panorama.*



FIG. 7. *GLC images created from a light field. Top row: a pencil, bilinear, and pushbroom image. Bottom row: an cross-slit, twisted orthographic, and orthographic image.*

where $[\sigma_1, \tau_1, 0, 0]$, $[\sigma_2, \tau_2, 1, 0]$, and $[\sigma_3, \tau_3, 0, 1]$ represent the three generator rays. We use the uv plane as the default image plane. For each pixel $[i, j]$, we compute the corresponding ray in the GLC as $r = [i, j, 0] + \lambda[\sigma_1 + i\sigma_2 + j\sigma_3, \tau_1 + i\tau_2 + j\tau_3, 1]$. We then trace r using the PovRay ray-tracing engine.

Our GLC-RT also supports more complicated GLC descriptions with an arbitrary collineation as:

```
camera { glc_camera
        Generator_Rays
```

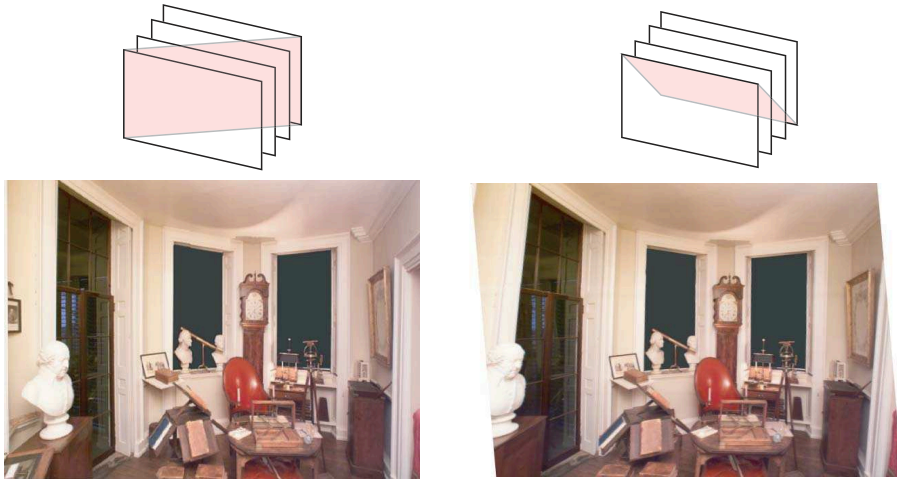


FIG. 8. *Synthesizing GLCs by cutting through a 3D horizontal light fields. Left: we cut through the light field vertical diagonally to generate a cross-slit GLC. Right: we cut through the light field vertical diagonally to generate a pencil GLC.*

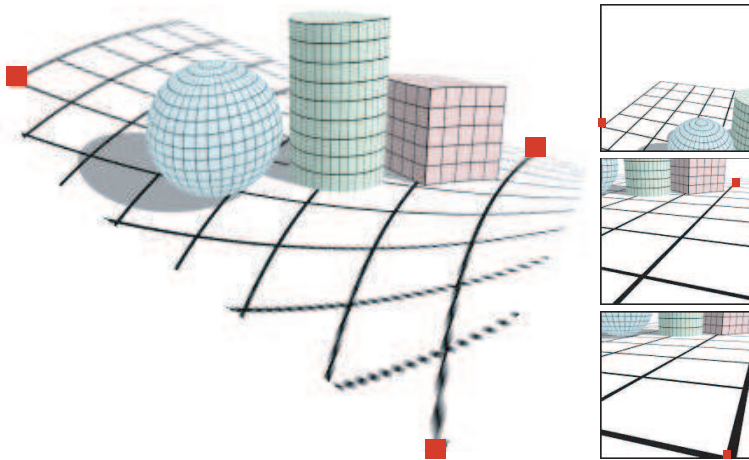


FIG. 9. *A bilinear GLC image synthesized from three pinhole cameras shown on the right. The generator rays are highlighted in red.*

$$\begin{aligned}
 & \langle o_1^x, o_1^y, o_1^z \rangle \quad \langle o_2^x, o_2^y, o_2^z \rangle \quad \langle o_3^x, o_3^y, o_3^z \rangle \\
 & \langle d_1^x, d_1^y, d_1^z \rangle \quad \langle d_2^x, d_2^y, d_2^z \rangle \quad \langle d_3^x, d_3^y, d_3^z \rangle \\
 & \text{Location } \langle L^x, L^y, L^z \rangle \\
 & \text{Up } \langle U^x, U^y, U^z \rangle \\
 & \text{Right } \langle R^x, R^y, R^z \rangle \\
 & \}
 \end{aligned}$$

where each generator ray r_i is represented by its origin o_i and direction d_i , for $i = 1, 2, 3$. The image plane (collineation) is specified by the center of the plane \hat{L} ,

the up vector \vec{U} , and the right vector \vec{R} .

In our implementation, we first warp each generator ray r_i to its default 2PP coordinate $r_i[u_i, v_i, s_i, t_i]$ by intersecting it with the $z = 0$ and $z = 1$ plane as:

$$(29) \quad \begin{aligned} u_i &= o_i^x - \frac{o_i^z}{d_i^z} d_i^x, v_i = o_i^y - \frac{o_i^z}{d_i^z} d_i^y, \\ s_i &= o_i^x + \frac{1 - o_i^z}{d_i^z} d_i^x, t_i = o_i^y + \frac{1 - o_i^z}{d_i^z} d_i^y, \quad i = 1, 2, 3 \end{aligned}$$

We then compute the GLC's canonical representation. Recall that the three canonical generator rays are also affine combinations of r_1 , r_2 , and r_3 . Therefore, we can solve for the affine coefficients α_i and β_i that satisfy

$$(30) \quad \begin{aligned} [\tilde{u}_i, \tilde{v}_i, \tilde{s}_i, \tilde{t}_i] &= \alpha_i[u_1, v_1, s_1, t_1] + \beta_i[u_2, v_2, s_2, t_2] \\ &+ (1 - \alpha_i - \beta_i)[u_3, v_3, s_3, t_3], \quad i = 1, 2, 3 \end{aligned}$$

where $[\tilde{u}_1, \tilde{v}_1] = [0, 0]$, $[\tilde{u}_2, \tilde{v}_2] = [1, 0]$, $[\tilde{u}_3, \tilde{v}_3] = [0, 1]$ as

$$(31) \quad \alpha_i = -\frac{-u_2\tilde{v}_i + u_3\tilde{v}_i + \tilde{u}_i v_2 - \tilde{u}_i v_3 + u_2 v_3}{u_2 v_1 - u_3 v_1 - u_1 v_2 + u_3 v_2 + u_1 v_3 - u_2 v_3}$$

$$(32) \quad \beta_i = -\frac{u_1\tilde{v}_i - u_3\tilde{v}_i - \tilde{u}_i v_1 + \tilde{u}_i v_3 - u_1 v_3}{u_2 v_1 - u_3 v_1 - u_1 v_2 + u_3 v_2 + u_1 v_3 - u_2 v_3}$$

By substituting α_i and β_i into Equation (30), we can compute the relative ray coordinate $[\tilde{\sigma}_i, \tilde{\tau}_i, \tilde{u}_i, \tilde{v}_i]$ as

$$(33) \quad \tilde{\sigma}_i = \tilde{s}_i - \tilde{u}_i, \quad \tilde{\tau}_i = \tilde{t}_i - \tilde{v}_i, \quad i = 1, 2, 3.$$

Assume the desired image resolution is $[w, h]$, to trace out a ray from each pixel $[i, j]$, we compute the actually 3D point \dot{P} that corresponds to pixel $[i, j]$ as

$$(34) \quad \dot{P} = \dot{L} + k_x \vec{R} + k_y \vec{U}$$

where

$$(35) \quad k_x = \frac{i}{w} - 0.5, \quad k_y = 0.5 - \frac{j}{h}$$

We then use the GLC Projection Equation (16) to find the ray that passes through \dot{P} . Finally, we trace the ray using the PovRay ray-tracing engine.

Notice that our implementation minimizes the modifications to the PovRay framework by changing the camera model engine and the ray-generator engine of PovRay, not the ray-tracing engine. Furthermore, antialiasing is generically implemented in our GLC-RT. This is because PovRay uses supersampling to reduce the aliasing artifacts. In our implementation, we simply compute the corresponding 3D points for subpixels using Equation (34) and determine the GLC projection of these points using Equation (16).

In Fig.4, we compare different GLC images ray-traced from a synthetic scene. The distortions of the curved isolines on the objects illustrate various multi-perspective effects of GLC cameras. For instance, all lines in 3D space are preserved as lines on the 2D image for orthographic and pinhole camera while they are curved in most other GLCs. In particular, lines are strongly twisted in the images of twisted orthographic and pencil camera. This is mainly because the rays collected by both cameras are twisting in 3D space in nature. Similar to pinhole cameras, many GLCs have "vanishing" points of parallel lines. Furthermore, in pushbroom camera, objects far away from the camera are stretched while their size is preserved under pinhole camera, orthographic, and cross-slit cameras, as has been analyzed in Section 4.

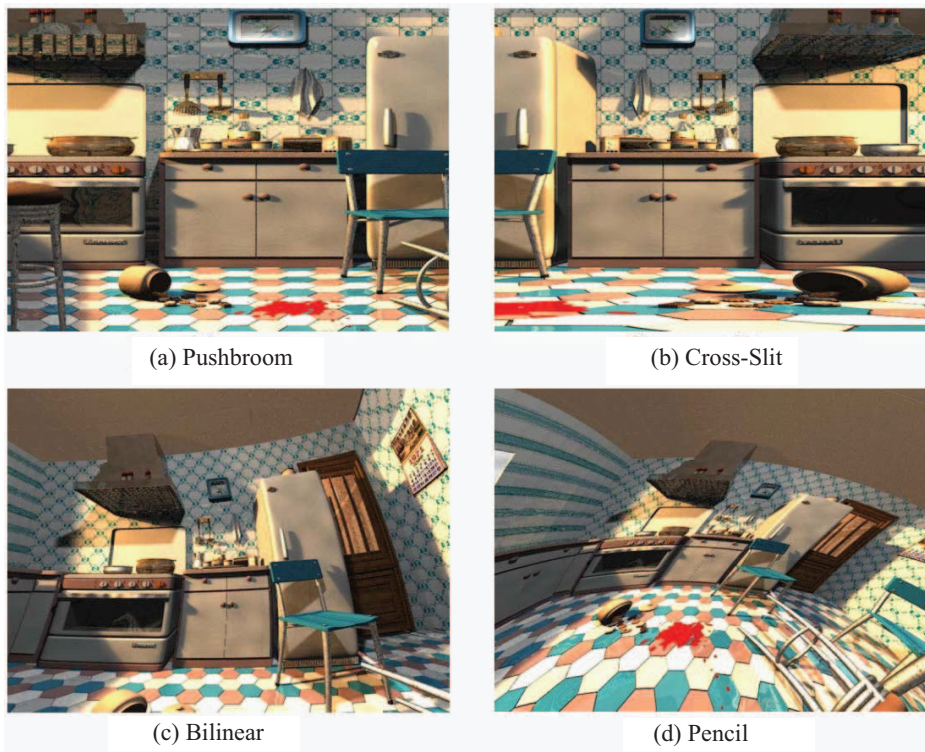


FIG. 10. *Kitchen scene rendered by GLC raytracer using pushbroom camera (a), cross-slit camera (b), bilinear camera (c), and pencil camera (d).*

In Fig.10, we render different GLC cameras in a complex kitchen scene. Despite their incongruity of view, these multiperspective images are still able to preserve spatial coherence while exhibiting unique distortions. In Fig.11, we compare a perspective panorama and a cross-slit panorama, both rendered using our GLC-RT in a city scene. The cross-slit panorama is able to illustrate smooth transitions from the east side of the city to the west side.

We can also generate non-perspective autostereoscopic images, as shown in Fig.12. While conventional autostereoscopic displays renders each hexagon lenslet as a pinhole

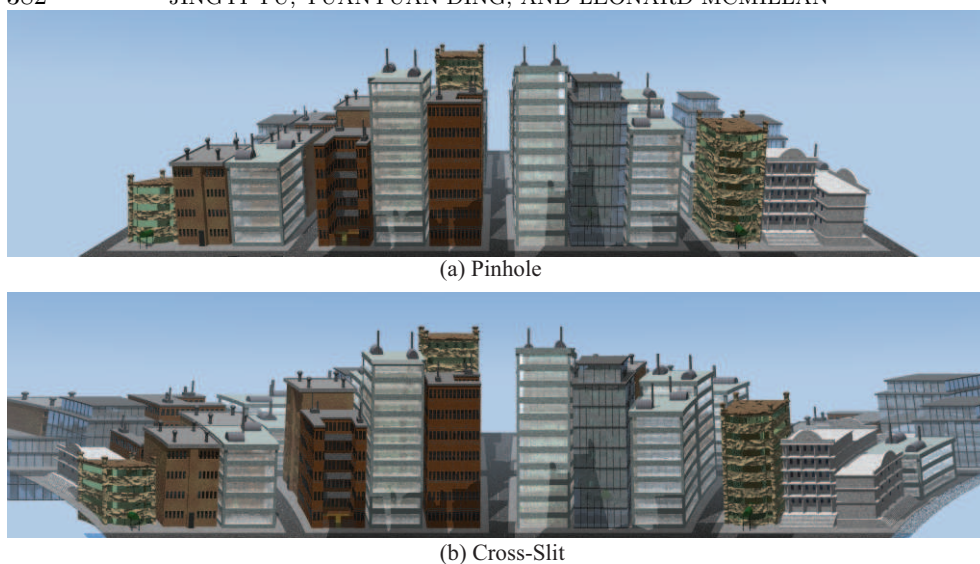


FIG. 11. *Panoramas of city scene rendered using (a) pinhole camera and (b) cross-slit camera.* camera, we render each hexagon as a pushbroom camera using our GLC-RT. This allows us to create "rotational motion parallax" in addition to the stereo parallax, as shown in the supplementary video.

8. Conclusions and Future Work. We have presented a General Linear Camera (GLC) model that unifies many previous camera models into a single representation. The GLC model is capable of describing all perspective (pinhole), orthographic, and many multiperspective (including pushbroom and two-slit) cameras, as well as epipolar plane images. It also includes three new and previously unexplored multiperspective linear cameras. The GLC model is both general and linear in the sense that, given any vector space where rays are represented as points, it describes all 2D affine subspaces (planes) that can be formed by affine combinations of 3 rays. The incident radiance seen along the rays found on subregions of these 2D affine subspaces are a precise definition of a projected image of a 3D scene.

We model the GLC imaging process in terms of two separate stages: the mapping of 3D geometry to rays and the sampling of those rays over an image plane. We have derived a closed-form solution to projecting 3D points in a scene to rays in a GLC and a notion of GLC collineation analogous to pinhole cameras. Finally, we have developed a multiperspective ray-tracer for the direct rendering of GLC images. The GLC ray-tracer is able to create a broad class of multiperspective effects and it provides flexible controls to reduce multiperspective distortions via collineations.

In the future, we plan to use the commodity graphics hardware to directly render GLC images directly from the scene geometry. The GLC projections of a line onto the GLC image plane is a quartic rational. Thus, it is possible to modify the rasterization unit or the shader to efficiently render the geometry directly into a GLC image using

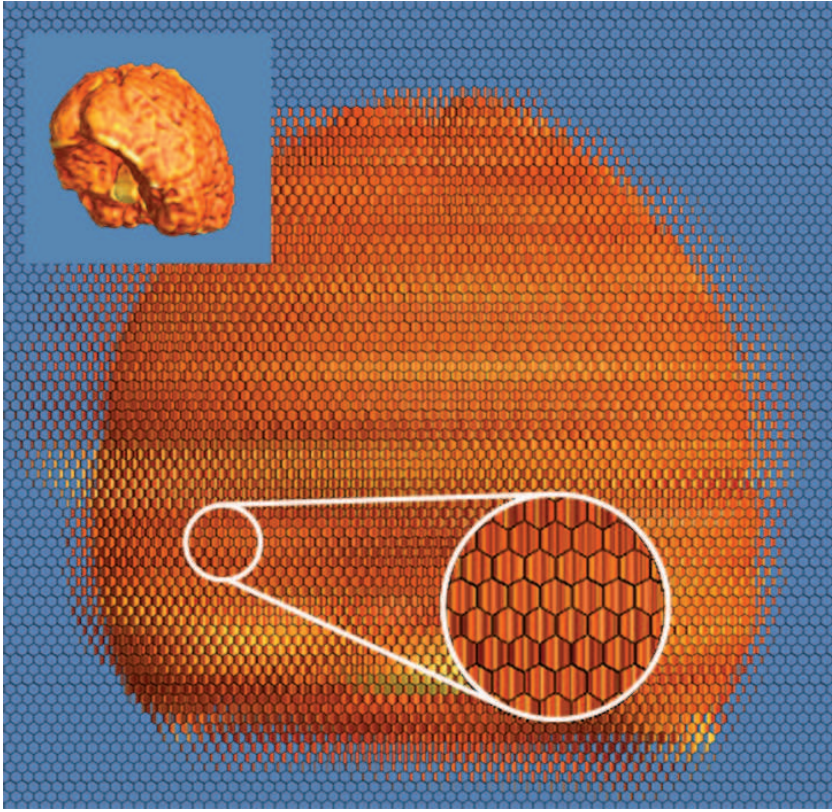


FIG. 12. *An autostereoscopic image rendered using GLC raytracer, each hexagon corresponds to a pushbroom camera.*

the graphics hardware. A real-time renderer will also be beneficial to the interactive design of multiperspective rendering, the creation of backdrops for cel-animation, and image-based animations.

Finally, we are also interested in designing actual GLC cameras and new stereo algorithms to enable 3D reconstructions from very wide field-of-view images, as well as other multiperspective images. We have recently shown [3] that most of the GLC models can form close-to-stereo image pairs. This makes the GLC model a promising tool for synthesizing multiperspective stereo fusions.

REFERENCES

- [1] ASEEM AGARWALA, MANEESH AGRAWALA, MICHAEL COHEN, DAVID SALESIN, AND RICHARD SZELISKI, *Photographing long scenes with multi-viewpoint panoramas*, ACM Transactions on Graphics (Proceedings of SIGGRAPH 2006), 2006.
- [2] R. C. BOLLES, H. H. BAKER, AND D. H. MARIMONT, *Epipolar-Plane Image Analysis: An Approach to Determining Structure from Motion*, International Journal of Computer Vision, Vol. 1 (1987).
- [3] YUANYUAN DING AND JINGYU YU, *Epsilon Stereo Pairs*, British Machine Vision Conference, 2007.

- [4] S. GORTLER, R. GRZESZCZUK, R. SZELISKI, AND M. COHEN, *The Lumigraph*, Proc. ACM SIGGRAPH '96, (1996), pp. 43–54.
- [5] XIANFENG GU, STEVEN J. GORTLER, AND MICHAEL F. COHEN, Polyhedral geometry and the two-plane parameterization. Eurographics Rendering Workshop 1997 (1997) pages 1-12.
- [6] R. GUPTA AND R.I. HARTLEY, *Linear Pushbroom Cameras*, IEEE Trans. Pattern Analysis and Machine Intelligence, 19:9(1997), pp. 963–975.
- [7] R.I. HARTLEY AND A. ZISSERMAN, *Multiple View Geometry in Computer Vision*, Cambridge Univ. Press, 2000.
- [8] R. KINGSLAKE, *Optics in Photography*. SPIE Optical Eng., Press, 1992.
- [9] M. LEVOY AND P. HANRAHAN, *Light Field Rendering*, Proc. ACM SIGGRAPH '96, (1996) pp. 31–42.
- [10] B. NEWHALL, *The History of Photography, from 1839 to the Present Day*, The Museum of Modern Art, (1964), pp. 162.
- [11] T. PAJDLA, *Stereo with Oblique Cameras*, Int'l J. Computer Vision, vol. 47, nos. 1/2/3 (2002), pp. 161–170.
- [12] T. PAJDLA, *Geometry of Two-Slit Camera*, Research Report CTU–CMP–2002–02, March 2002.
- [13] S. PELEG, M. BEN-EZRA, AND Y. PRITCH, *Omnistereo: Panoramic Stereo Imaging*, IEEE Trans. Pattern Analysis and Machine Intelligence, 23:3(2001), pp. 279–290.
- [14] P. RADEMACHER AND G. BISHOP, *Multiple-center-of-Projection Images*, Proc. ACM SIGGRAPH '98, (1998), pp. 199–206.
- [15] AUGUSTO ROMN, GAURAV GARG, AND MARC LEVOY, *Interactive Design of Multi-Perspective Images for Visualizing Urban Landscapes*, IEEE Visualization, 2004.
- [16] S. M. SEITZ, *The Space of All Stereo Images*, Proc. Int'l Conf. Computer Vision '01, vol. I, (2001), pp. 26–33.
- [17] J. SEMPLE AND G. KNEEBONE, *Algebraic Projective Geometry*, Oxford: Clarendon Press, 1998.
- [18] H.-Y. SHUM, A. KALAI, AND S. M. SEITZ, *Omnivergent stereo*, in: Proc. 7th Int. Conf. on Computer Vision, (1999) pp. 22–29.
- [19] D. SOMMERVILLE, *Analytical Geometry of Three Dimensions*, Cambridge University Press, 1959.
- [20] T. TAKAHASHI, H. KAWASAKI, K. IKEUCHI, AND M. SAKAUCHI, *Arbitrary View Position and Direction Rendering for Large-Scale Scenes*, Proc. IEEE Conf. Computer Vision and Pattern Recognition, (2000), pp. 296–303.
- [21] D. WOOD, A. FINKELSTEIN, J. HUGHES, C. THAYER, AND D. SALESIN, *Multiperspective Panoramas for Cel Animation*, Proc. ACM SIGGRAPH '97, (1997), pp. 243-250.
- [22] J. Y. ZHENG AND S. TSUJI, *Panoramic Representation for Route Recognition by a Mobile Robot*, Int'l J. Computer Vision, 9:1(1992), pp. 55–76.
- [23] A. ZOMET, D. FELDMAN, S. PELEG, AND D. WEINSHALL, *Mosaicing New Views: The Crossed-Slits Projection*, IEEE Trans. on PAMI, (2003), pp. 741–754.
- [24] POV-Ray: The Persistence of Vision Raytracer. <http://www.povray.org/>