Polar-vision[1]: A Novel Collinearity Equation of Perspective Projection in Polar Coordinate System

Zhengkang Zuo[1, 2, 3, *], Bin Zhang[1]

[1] Working Group of Polar-vision, Taiyuan University of Technology, Taiyuan, 030024, China
[2] School of Earth and Space Science, Peking University, Beijing, 100871, China
[3] DASPATIAL Co., Ltd, Wuhan, 430000, China

Abstract: Progress has been made in the community of photogrammetry and 3d computer vision in addressing the mathematical challenge posed by the collinearity equation. We introduce a new method for establishing the coordinate reference for 2d pixels and 3d landmarks using 'angular coordinates'. The mathematical relationships required for converting 3d landmarks, expressed in angular coordinates, to the camera framework are presented. The landmarks are then projected using perspective projection to obtain 2d pixels represented in angular coordinates. This framework is formally nominated as the 'Polar-vision[1]', which has been developed and integrated into the commercial software G3D-Cluster. Its application to pinhole camera image processing has demonstrated superior efficiency and admission rates of tie points, as well as reconstruction detail capabilities, compared to OpenMVG, achieving approximately a 1.4x improvement. The project 'Key Technologies and Tool System for Realistic 3D Modeling through Integration of Multi-Source Information in the Space-Air-Ground Domain' was awarded First Prize at the 2023 Surveying Science and Technology Awards, with Polar-vision[1] as one of the innovative points.

Keywords: Collinearity Equation, Perspective Projection, Angular Parameterization, Polar Coordinate System.

1. Introduction

1.1 Motivation

The collinearity equation, also known as the projection matrix (Hartley & Zisserman, 2003) in 3d computer vision, is a fundamental theory in Photogrammetry. It serves to connect 2d images with the 3d physical world, allowing us to extract and optimize 3d metric information from 2d images using various technologies such as camera calibration (Zhang, 1998), structure from motion (Schonberger & Frahm, 2016), and multi-view stereo (Goesele et al., 2006). It is important to note that the well-known collinearity equation conveys the concept of Euclidean geometry (Ryan, 1986), describing a bundle of rays that connect the center of projection, 2d pixels (image points), and 3d landmarks (object points) determined by the Euclidean coordinate.

In reality, a bundle of optical rays is better represented in the formula of polar coordinate due to its directional property. Instead of using mutually perpendicular axes, angles are more suitable for describing the vector nature of hitting positions of optical rays. The hitting points of an optical ray on the focal plane and the object are referred to as 2d pixel and 3d landmark, respectively (Rosenfeld, 1988). This inspires us to redefine those 2d-3d points and their mathematical relationship using the concept of "angular coordinate". That is a novel principle of collinearity equation to serve the community of Photogrammetry, nominated as *Polar-vision[1]*.

1.2 Related Works

The collinearity equation consists of three components that can be parameterized: pixel, landmark, and rotation. By optimizing these components non-linearly across different spatial domains, it has a significant impact on the accuracy of calculations related to camera distortions, interior-exterior orientation parameters, and the quality of 3D reconstruction (Triggs et al., 2000).

1.2.1 Pixel

In situations where GPS, IMU, and laser altimeters are unavailable, 2d pixels basically serve as the only observation constraint for photogrammetry. According to research, the majority of work in the fields of photogrammetry and 3d computer vision parameterizes the position of pixels into Euclidean coordinates (x, y). It estimates all camera states during photogrammetry by constructing observation equations for x and y and reconstructs 3D sparse scenes (Li, 1995; Mikhail and Bethel, 2001).

The camera lens deviates from the ideal pinhole model, causing light to refract through lenses of various shapes. This refraction results in the "bending" of light rays (though this term is used figuratively for ease of understanding, not to imply physical bending). Consequently, the pixels calculated by the observation equation represent ideal positions, disregarding distortion effects caused by lens imperfections. When using pixels with distortion effects in photos as observation constraints to estimate the camera state, significant systematic errors may be introduced in the estimation results. Therefore, it is necessary to model the distortion effect to accurately parameterize the pixels in the photo. In the 20th century, the Brown Conrady model was widely regarded as the "standard" radial and tangential distortion model (Brown, 1996; Conrady, 1919). However, its inconsistency between the Euclidean parameterization of pixel coordinates and the polar forms of distortion effects (radial and tangential) rendered it incapable of adapting to the complex distortion effects of wide-angle, ultra-wide-angle, and fisheye lenses. It wasn't until the early 21st century that a breakthrough occurred with the introduction of the Kannala Brandt model, published in the TPAMI journal (Kannala, Brandt, 2006). This model marked the first successful replacement of the Brown Conrady model in a century. It characterized the distortion effect as a function of the incident angle of light passing through the lens. Some key advantages of this model include: (1) The parameterized distortion function based on the incident angle (denoted as θ) is smoother and more amenable to modeling as a power series. (2) The formula can be adjusted according to θ to support various types of projections.

In simpler terms, photogrammetry begins with the observations captured in photos and then reconstructs the process of translating a 3d scene into a 2d photo through collinearity equations. The goal is to recover the depth information lost in the 2d photo. This translation process is known as projection, where the 3d scene is rigidly transformed into the camera frame and then projected onto the photo following specific rules. This forms the underlying construction logic of collinearity equations. The projection rules supported by the Kannala-Brandt model are as follows: (1) Perspective: θ is parameterized as $\tan^{-1}$ (r/f) (Yang et al., 2005). (2) Stereographic: θ is parameterized as $2\tan^{-1}$ (r/2f) (Chang et al., 2013). (3) Equidistant: θ is parameterized as r/f (Hughes et al., 2010). (4) Equisolid: θ is parameterized as $2\sin^{-1}$ (r/2f) (Eichenseer et al., 2015). (5) Orthogonal: θ is parameterized as $\sin^{-1}$ (r/f) (Zhang et al., 2015).

Inspired from (1) the Kannala-Brandt model's parameterization design of distortion effects, and (2) a method presented at ICCV conference that uses 'angular coordinates' to parameterize reprojection errors (Lee and Civera, 2019). we propose an angular parameterization method for pixel coordinates. This involves replacing 'Euclidean coordinates' with 'angular coordinates' to parameterize the position of 2d pixels, thus adapting to the polar coordinate form of distortion effects and the angular distance definition of reprojection errors. At this stage, the camera state and the 3d scene are no longer restricted by the x and y observation equations. Instead, they are limited by a new set of angular coordinate constraints. The observation equations may benefit from the unique properties of angular coordinates. Unlike Euclidean coordinates, which are defined in two specific directions of the photograph, angular coordinates exhibit isotropy and are independent of photograph orientation. Moreover, compared to the Euclidean distances (d_0, d_1), the distance between two pixels measured in 'angular coordinates' (θ_0, θ_1) exhibits intrinsic rotational invariance. This property makes the objective function defined by it more robust against outliers encountered during the optimization process.

1.2.2 Landmark

According to research, almost all studies (Wang, 2007; Zhang, 2007; Liu, 2013; Wang, 2016) in the field of photogrammetry parameterise the positions of 3d landmarks in Euclidean coordinates (X, Y, Z). However, this parameterisation is no longer sufficient to address certain specific 3d tasks in the field of computer vision. We focus mainly on monocular vision (Mur Artal et al., 2015) and does not currently address SLAM tasks based on binocular vision (Mur Artal and Tardós, 2017) and LiDAR (Cole and Newman, 2006). In the absence of range sensors, bearing only localisation and mapping (BOLAM) tasks based on monocular vision assume that the target moves linearly in a Euclidean coordinate system. However, the orientation information of the target observed by the sensors is expressed in polar coordinates. Continuing with conventional parameterisation would introduce significant nonlinear challenges into the observation equation (Deans, 2005). In addition, it would prevent BOLAM systems based on Extended Kalman Filter (EKF) from achieving real-time (no delay) initialisation mapping (Chiuso et al., 2002; Bailey, 2003).

The initial mapping phase is crucial and challenging for BOLAM tasks. Monocular vision provides an irreversible rank-deficient measurement, making it difficult to estimate camera trajectories and scene maps through subsequent EKF filtering. There are two main types of methods: (1) Delayed initialization occurs when the camera motion acquires sufficient parallax before initialization begins. During this motion, low-parallax observations are entirely disregarded, leading to inconsistency in the mapping process and potentially resulting in filter failure (Davison, 2003). (2) Undelayed Landmark Initialization (ULI) involves initializing points either in the direction of camera motion or located very far from the camera upon the first observation. These landmarks may remain visible throughout the entire motion process. ULI is employed to constrain the camera direction and enhance consistency throughout the mapping process (Sola et al., 2005; 2008).

If the landmarks utilized by ULI are represented as Euclidean points, the uncertainty interval of their depth information (Z-coordinate) remains unbounded and cannot be resolved using EKF (Terejanu, 2008). At the 2008 ICCV conference, a novel parameterization method was proposed, which involves parameterizing the depth of landmarks (unobservable degrees of freedom) into inverse depth. This approach transforms the uncertainty interval from unbounded to bounded (Civera et al., 2008). Another parameterization method, similar to inverse depth but distinct from it, is inverse distance. Inverse distance is defined as the reciprocal of the distance to the origin of the world reference frame (Sola et al., 2012), and it is expressed in a homogeneous manner as (**m**, ρ). Note that when transformed into the camera frame, the inverse distance becomes the reciprocal of the distance to the camera center. Compared to Euclidean points, while the uncertainty interval of the inverse distance point is bounded, bilinearity is introduced into the equations when transformed into the camera framework. This introduces some unfavorable factors to the performance of the EKF, which requires a reasonably linear system (Hartley and Zisserman, 2003). Therefore, inspired by the reference (Eade and Drummond, 2006), the concept of anchor (the initial camera center) was introduced to alleviate the adverse effects of bilinearity. Compared to the inverse depth defined by three specific directions in space, the inverse distance has the advantage of isotropy and is independent of the direction of the spatial reference frame. It is expressed as (**p**$_0$, **m**, ρ). Since **m** represents the directional vector of landmarks relative to anchor points, expressing it as Euclidean coordinates is clearly redundant. Therefore, the direction vector is parameterized as the azimuth and elevation angles of the landmarks relative to the anchor. The homogeneous expression of the inverse distance is then refined to (**p**$_0$, α, β, ρ). Compared to Euclidean points, the use of inverse distance parameterization leads to improved consistency in EKF-SLAM mapping based on monocular vision. It is important to note that EKF is not the only option for SLAM systems. However, while inverse distance is tailored to EKF, it exhibits poor performance in Bundle Adjustment (BA). In BA, landmarks in the direction of camera motion can result in ill-conditioned normal equations (Konolige and Agrawal, 2008).

To address the issue of inverse depth in BA, (Zhao et al., 2015) proposed a parameterization of landmark depth using parallax angle defined by two anchors, expressed homogeneously as (**p**$_m$, **p**$_a$, α, β, γ). Their open-source project surpasses the performance of SBA (Lourakis and Argyros, 2009) and sSBA (Konolige and Garage, 2010) projects, which use Euclidean points. (Zhao et al., 2015) analysed the effects of parallax angle, inverse depth, and Euclidean parameterisation of landmarks on BA performance in a 2d BOLAM system. In contrast, (Zuo et al., 2023) investigated the impact of parallax angle parameterisation of landmarks on BA performance in a 3d BOLAM system. (Sun, 2015) extended the parameterisation of parallax angle from the BOLAM system to the UAV photogrammetry based on the pinhole camera model. The IEEE standard for this method was published in 2023 (IEEE 1937.11-2023). Additionally, (Zuo,

2023) applied the parameterization of parallax angle to satellite photogrammetry using the linear pushbroom model. All of these investigations have shown a significant decrease in the correlation coefficients of 3d structural variables parameterized by parallax angles. This reduction significantly relaxes the strict geometric requirements imposed by adjustment models in photogrammetric measurements.

However, the pixel coordinates are still represented in the Euclidean domain, resulting in a semi-angular domain transformation from the landmark to the pixel. This causes inherent dimensional inconsistency between the observational constraints and the state estimations, making the rigid transformation difficult and complicating the collinearity equation. As a result, numerical instabilities and suboptimal convergence properties occur during the adjustment and optimization processes. Therefore, we parametrize the pixel coordinates and the rigid transformation in the angular domain.

1.2.3 Rotation

The concept of rotation is typically classified into two categories. The first involves rotating a target object within a fixed spatial reference frame, such as the movements of the seven joints of a robot (ankle, knee, hip, shoulder, elbow, wrist, finger) or the rolling, pitching, and yawing actions of an aircraft. The second category involves rotating the spatial reference frame around a fixed target object (Foley, 1996).

In photogrammetry, data collection usually involves rotating the aircraft and camera gimbals. Data process, however, involves rotating a spatial reference frame. We focus solely on the latter type of rotation, which is the initial step in establishing a connection between the physical and digital worlds.

Euler angles were first proposed by (Euler, 1765). The concept was originally presented in Latin and can be referenced in (Weisstein, 2009). Any rotation of a rigid body can be parameterized by rotating three angles around three orthogonal axes (θ_x, θ_y, θ_z) in a specific order. Therefore, it is necessary to clarify the rotation sequence of the three axes before using Euler angles. However, this introduces the problem of universal deadlock, as noted by (Brezov et al., 2013).

In 1840, French mathematicians (Rodrigues, 1840) proposed the concept of the Axis angle parameterization to represent rotation as ($\mathbf{n}$, θ). The original version was in French and can be referred to (Gray, 1980; Caccavalle et al., 1999). The Rodrigues rotation formula, described by (Murray et al., 2017), can rotate any 3d vector around the unit axis $\mathbf{n}$ by an angle θ. This formula was also adopted by the OpenMVG framework, as noted by (Moulon et al., 2017). The angle θ can be decomposed into three orthogonal directions of the rotation axis $\mathbf{n}$, similar to the Euler angle rotation around three orthogonal axes. However, unlike Euler angles, the Rodrigues rotation formula does not require a predefined rotation sequence, which mitigates the issue of gimbal lock. (Craig, 2005) proposed a parameterization method for the rotation vector by multiplying the angle θ by the axis $\mathbf{n}$. This method uses a 3d vector to represent the rotation axis and angle. The OpenMVG framework does not directly involve the axis angle in gradient calculation. Instead, it indirectly updates the axis angle by computing the partial derivative of the observation equation with respect to the rotation vector. The landmark is then rotated to the camera frame using the Rodrigues formula.

In 1840, the Irish mathematician Hamilton introduced the concept of Quaternions (Hamilton, 1840), which parameterizes rotations into a pair of conjugate quaternions (q, q^*). To rotate any 3d vector $\mathbf{v}$ around the unit axis $\mathbf{n}$ by angle θ, it is written as a pure quaternion v, left-multiply by q and right-multiplied by q^*. This concept is similar to the axis-angle representation, as conjugate quaternions can be transformed equivalently using the Rodrigues formula. Quaternions are not used directly in gradient calculations. Instead, they are first converted to axis angles and then to rotation vectors to compute correction increments. These updated rotation vectors are subsequently converted back to quaternions. The landmarks are transformed to the camera frame by multiplying them with conjugate quaternions.

Rotation matrix (Weisstein, 2003) is necessary to directly rotate landmarks to the camera frame, as Euler angles cannot achieve this. The rotation matrix shares similarities with the Rodrigues formula and conjugate quaternions. It was first developed based on research conducted by the French mathematician (Cauchy, 1815). Deriving the rotation matrix directly to estimate the camera rotation is not possible due to the addition and subtraction operations involved in gradient calculations, which lack closure. To address this issue, Lie group and Lie algebra theory (Lie, 1880; 1888) can be introduced. For further details, please refer to the original work in French (Sola et al., 2018).

Indeed, these rotation methods are developed in the Euclidean space domain, which is not suitable for rotating landmarks in the angular domain. Therefore, we propose a new method to rotate landmarks in the full angular domain.

2. Methods

2.1 Parameterization for Pixel in Angular Domain

We define two angles, namely the viewing-angle and the polar-angle, to parameterize the positions of pixels, instead of using Euclidean coordinates (x, y). The viewing-angle θ is defined as the angle between the ray passing through the pixel and the principal axis, which is equivalent to the distance of the pixel from a pole. The polar angle φ represents the angle of the pixel from the pole axis. As shown in Figure 1, homonymous pixels $\mathbf{x}_i$ and $\mathbf{x}_j$ are parametrized in angular domain as

$$\mathbf{x}_i = \begin{pmatrix} \theta_i \\ \varphi_i \end{pmatrix} ; \; \mathbf{x}_j = \begin{pmatrix} \theta_j \\ \varphi_j \end{pmatrix}, \tag{1}$$
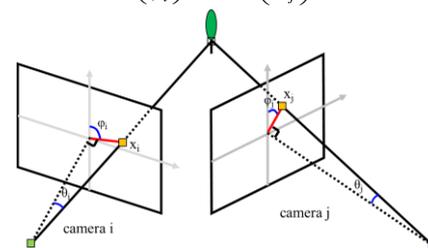
Figure 1. Concept of pixel in angular domain.

2.2 Parameterization for Transformation in Angular Domain

In the angular domain, the transformation from landmarks parameterized by three angles (α, β, γ) to pixels parameterized by two angles (θ, φ) is defined as

$$\begin{pmatrix} \theta_i \\ \varphi_i \end{pmatrix} = \Gamma_1 \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} ; \; \begin{pmatrix} \theta_j \\ \varphi_j \end{pmatrix} = \Gamma_2 \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}, \tag{2}$$

As shown in Figure 2, the left camera is a main anchor, and the right is an associate anchor. Γ_1 and Γ_2 represent the mapping relationships from the landmark (α, β, γ) to homonymous pixels (θ_i, φ_i) and (θ_j, φ_j) respectively.
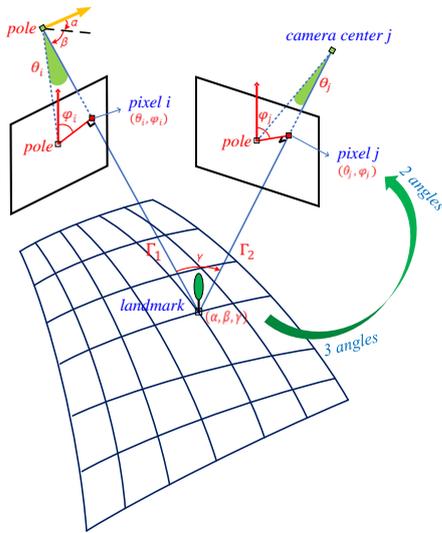
Figure 2. Concept of transformation ($\alpha\beta\gamma\rightarrow\theta\varphi$) in angular domain.

2.3 Geometric Relationship between Pixel and 3D Rotation in Angular Domain

As shown in Figure 3, we can get the geometric relationship between the pixel (θ, φ) and the 3D rotation (e_y, e_x, e_z) by rotating the plane (that the landmark lies on) in the order of $e_y \rightarrow e_x \rightarrow e_z$ to the plane (that the pixel lies on). It can be observed that the construction of the plane (that the pixel lies on) only requires twice rotations ($e_y \rightarrow e_x$) from the plane (that the landmark lies on), as shown in Figure 3a. Then, this plane is rotated by an angle e_z to form the final image plane, as shown in Figure 3b. This is a significant difference between our method and traditional methods that use Euclidean coordinates (x, y) to parameterize the pixel.
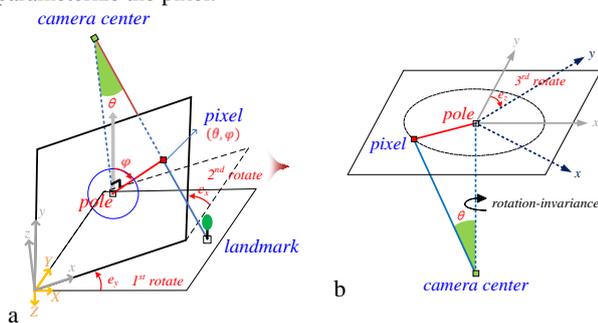
Figure 3. Geometric relationship between the pixel (θ, φ) and 3D rotation ($e_y\rightarrow e_x\rightarrow e_z$) in angular domain.

2.4 Derivative for View-angle Observation Equation in Angular Domain

Based on Figure 4, the observation from the view-angle θ is finally derived as

$$\theta = \sin^{-1}\left[\cos e \sin \beta + \sigma_2 \sin e \cos \beta \cos(\sigma_1 v + \alpha)\right], \quad (3)$$

where
$\quad \theta$ = view-angle of the pixel p
$\quad \alpha, \beta$ = azimuth and elevation angle of the landmark P

e_y, e_x = two rotation angles from the plane (that the landmark P lies on) to the plane (that the pixel p lies on).
e, v = functions of e_y and e_x

$$e = \cos^{-1}\left(\cos e_y \cos e_x\right), \quad (4)$$

$$v = \sin^{-1}\left(\tan e_y / \tan e\right), \quad (5)$$

σ_1, σ_2 = two constants

$$\sigma_1 = \begin{cases} 0 & e_y = 0 \\ -1 & e_x = 0 \\ -e_y e_x / |e_y e_x| & e_y e_x \neq 0 \end{cases}, \quad (6)$$

$$\sigma_2 = \begin{cases} -e_y / |e_y| & e_x = 0 \\ -e_x / |e_x| & otherwise \end{cases}, \quad (7)$$

2.4.1 Derivative in Angular Domain

We use Figure 4 to derive the Equation 3-5. Firstly, we give some notations as following:
(1) Vertex O, p and P are the camera center, the pixel and the landmark, respectively.
(2) Segment O-o and O-p are the main optical axis of the camera and the ray passing through the pixel p, respectively.

Secondly, we indicate some geometric relationships in this figure:
(1) Angle τ is the complementary angle of angle β.
(2) Angle u is the dihedral angle between the plane SAC and the plane OAP.
(3) The dihedral angle between the plane SBA and the plane SBC is the right angle.
(4) Segment S-C is parallel to O-o. Segment O-E is parallel to A-P. Segment O-F is parallel to A-D and G-H as well as B-C.
(5) Segment O-A is not parallel to G-K.
(6) Angle DAB is equal to α. Angle ACB is equal to v. Angle ASC is equal to e. Angle ABC is the right angle.
(7) Distance of the segment A-B is $\tan(e_y)$. Distance of the segment A-C is $\tan(e)$.

Finally, based on the formula for the solid angle of a tetrahedron, we derive the Equation 3 as

$$\cos \theta = \cos e \cos \tau + \sin e \sin \tau \cos u, \quad (8)$$

We derive the Equation 4 as

$$\cos e = \cos e_y \cos e_x + \sin e_y \sin e_x \cos(\pi / 2), \quad (9)$$

Based on the 7th geometric relationship that we indicate in Figure 4, we derive the Equation 5 as

$$\sin v = \tan e_y / \tan e \quad (10)$$
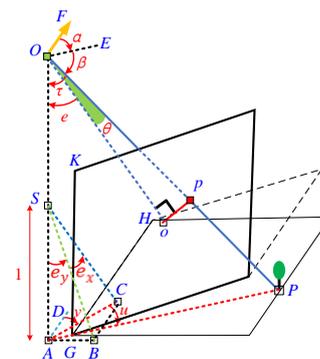
Figure 4. Derivative for view-angle observation equation.

3. Experiments

3.1 Comparative Method

The Open Multiple View Geometry (OpenMVG) is an internationally renowned open-source C++ framework for Structure from Motion (SfM) (https://github.com/openMVG). We utilize it as a comparative method to validate the accuracy and efficiency of our method in processing the pinhole imaging datasets.

3.2 Satellite Dataset

We utilized data from the Rosetta OSIRIS (Optical, Spectroscopic, and Infrared Remote Imaging System) released by the Max Planck Institute for Solar System Research, sourced from the European Space Agency's Rosetta Mission. The data was acquired using a frame CCD (Charge-Coupled Device) reflecting telescope mounted on the spacecraft, as shown in Figure 5a. This mission marks the first controlled landing of a lander on the surface of a comet, with the target comet being 67P/Churyumov-Gerasimenko, as shown in Figure 5b.
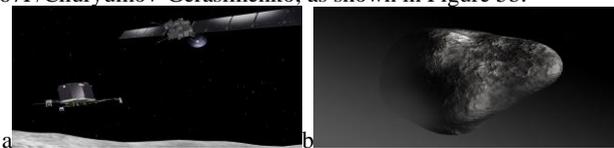
Figure 5. Rosetta mission (Image source:
https://www.open.ac.uk/science/research/rosetta/mission,
Copyright: ESA-J. Huart, 2013).

3.3 Results

We used 631 comet images, as shown in Figure 6a, to estimate the photographic orbit composed of satellite positions at different times. The resulting orbit is shown in Figure 6b.
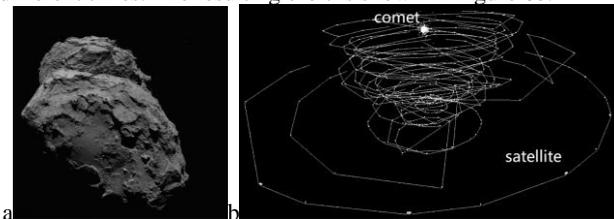
Figure 6. Photographic orbit constructed from 631 images.

Table 1 compares the processing accuracy and efficiency of OpenMVG and ours on this dataset. The evaluation metrics are the average reprojection error of all tie points (measured in pixels, denoted as "px") and the total runtime of the incremental SfM (measured in minutes, denoted as "m").

Indicators	OpenMVG	**Ours**	Improving
Accuracy (px)	0.482	**0.481**	Equivalent
Efficiency (m)	77	**56**	↑ 21 m (×1.4)
#Tie point	200,355	**200,708**	↑ 353
#Reprojected px	1,086,356	**1,089,739**	↑ 3,383

Table 1. Comparison on Rosetta dataset

It can be observed that on this dataset, our method achieves processing accuracy comparable to OpenMVG, both converging to the sub-pixel level (0.48 px). However, our method reduces the processing time by 21 minutes compared to OpenMVG (an improvement of about 1.4 times). Additionally, our method computes more tie points (an additional 353) and reprojected pixels (an additional 3,383) than OpenMVG, enhancing the admission rate of tie points and reconstruction detail capabilities.

Moreover, our method demonstrates higher efficiency than OpenMVG even with an increase in observation equations, validating the effectiveness and efficiency of our method, particularly when applied to satellite pinhole imaging models.

Furthermore, the result of the 3D model of the comet constructed using our methods is shown in Figure 7. Figure 7a displays a sparse point cloud resulting from the forward intersection of 200,708 tie points. Figures 7b and 7c show a dense point cloud and a mesh model, respectively.
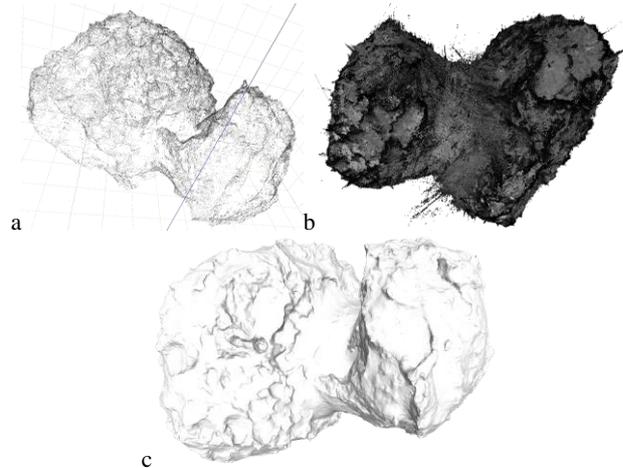
Figure 7. (a) Sparse point cloud (b) Dense point cloud (c) Mesh model of the comet constructed from 631 images.

3.4 UAV Datasets

We used data collected from the Unmanned Aerial Vehicle (UAV), which can be downloaded from the following websites: https://dashboard.aerosurvey.co.nz/files/shared/245.tar.gz and https://drive.google.com/u/0/uc?id=1Spu1F713Tw-z1XMdnrlD6NT4EhhFy2Lj&export=download.

3.5 Results

We used 857 images from the Poets_park_flats dataset, as shown in Figure 8a, to construct the sparse point cloud using our method. The resulting point cloud is shown in Figures 8b and 8c from different perspectives, which is from the forward intersection of 2,728,281 tie points.
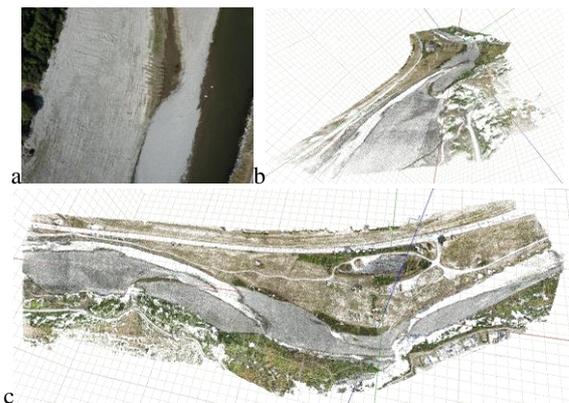
Figure 8. Sparse point cloud constructed from 857 images.

It can be seen from Table 2 that our method achieves a processing accuracy that is comparable to that of OpenMVG,

with both converging to the sub-pixel level (0.92 px vs. 0.93 px). Compared to OpenMVG, however, our method reduces the processing time by 444.2 minutes. It also computes more tie points (an additional 3,666) and reprojected pixels (an additional 14,648) than OpenMVG, improving the tie point acceptance rate and reconstruction detail capabilities.

Indicators	OpenMVG	**Ours**	Improving
Accuracy (px)	0.93	**0.92**	Equivalent
Efficiency (m)	1688.9	**1244.7**	↑ 444.2 m
#Tie point	2,724,615	**2,728,281**	↑ 3,666
#Reprojected px	12,026,356	**12,041,004**	↑ 14,648

Table 2. Comparison on Poets_park_flats dataset

We also constructed the sparse point cloud using our method with 493 images from the MapPilot dataset, as shown in Figure 9a. The resulting point cloud is shown in Figures 9b and 9c from different perspectives. These perspectives are from the forward intersection of 671,331 tie points.
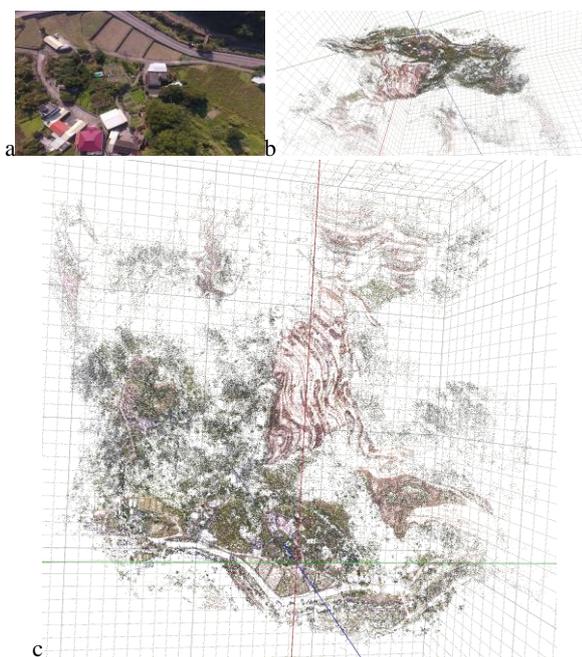
Figure 9. Sparse point cloud constructed from 493 images.

Table 3 shows that our method achieves processing accuracy comparable to OpenMVG, both converging to the pixel level (1.01 px). However, our method reduces processing time by 38.7 minutes compared to OpenMVG. It also computes 976 more tie points and 5,328 more reprojected pixels than OpenMVG, improving tie point acceptance rate and reconstruction detail capabilities.

Indicators	OpenMVG	**Ours**	Improving
Accuracy (px)	1.01	**1.01**	Equivalent
Efficiency (m)	164.1	**125.4**	↑ 38.7 m
#Tie point	670,355	**671,331**	↑ 976
#Reprojected px	1,760,356	**1,765,684**	↑ 5,328

Table 3. Comparison on MapPilot dataset

Therefore, the effectiveness and efficiency of our method, especially when applied to UAV pinhole imagery models, is confirmed by the fact that our method is more efficient than OpenMVG even with an increase in the number of observation equations.

4. Discussion

4.1 Potential Significance of *Polar-vision[1]*

In the proposed theory of *Polar-vision[1]*,
(1) the formula of angular coordinate (viewing angle, polar angle) is better to represent (modeling) position of 2d pixel with distortion in radial and tangential directions that are often associated with polar coordinates or circular motion.
(2) the numerical differences of angular coordinate of 2d pixels are much less than Euclidean coordinate, improving the stability of numerical computation;
(3) the formula of angular coordinate (azimuth angle, elevation angle, parallax angle) is better to represent and be used to estimate the depth of 3d landmark *via* the 3rd element in the view of narrow parallax angle;
(4) 2d pixel and 3d landmark are both represented in angular coordinate, consistent with attitude angles (roll, pitch, yaw) of the camera in the unit of measurement, degree or arc. This improves the stability of numerical computation in bundle adjustment to refine camera poses and positions of 3d landmark simultaneously;
(5) the mathematical relationship between 2d pixel and 3d landmark in angular coordinate is more concise and clearer to reveal the geometry of perspective projection and transformation to camera frame;
(6) Unlike *x* or *y* coordinate, which is defined with respect to a particular direction in image space, viewing angle has the advantage of being isotropic, that is, its properties are independent of the orientation of the image;
(7) The mathematical relationship between the "viewing angle" coordinate of 2d pixel and angular coordinate of 3d landmark is independent of the yaw attitude of the camera. That allows us define the observation equation for the "viewing angle" coordinate of the 2d pixel to refine the roll and pitch attitude of the camera, which makes the block of unknown parameters shrinks for more effective estimation of camera attitudes.

5. Conclusions

5.1 Contributions

The main contributions of this work are
(1) the parameterization of 2d-3d points *via* the angular coordinate and
(2) the construction of their mathematical relationship in polar coordinate system;

We nominate these two items of work as *Polar-vision[1]*, in which the superscript *[1]* reveals a series of potential works to do. That is to regain most theories of 3d computer vision in the perspective of polar coordinate system to exploit their potential significances so as to solve hard problems rooted in the primitive theories based on the Euclidean coordinate system. We will nominate those future works as *Polar-vision[2], Polar-vision[3]*, etc.

5.2 Limitations

There are several limitations in this work:
(1) The derivative for the polar-angle observation equation in the angular domain is more challenging than for the view-angle equation. Thus, we only add the view-angle observation equation as the additional constraint for the ParallaxBA project.

(2) The mathematical proof for the performance of the ParallaxBA project in 3d photographic geometry is currently unclear.

5.3 Future Works

Polar-vision[1] represents a novel collinearity equation formulated *via* angular coordinates to connect 2d pixels and 3d landmarks. Then, *Polar-vision[1]* as a basic equation is replaced of the Euclidean coordinate version in the framework of structure from motion. That serves the pinhole camera model.

In the basis of *Polar-vision[1]*,
(1) *Polar-vision[2]* will explore the potential significance of distortion modeling *via* angular coordinate of 2d pixel to serve the technology of camera calibration with distortion.
(2) *Polar-vision[3]* will investigate the advantage of depth estimation *via* angular coordinate of 3d landmark to serve the linear pushbroom camera model.

Inspired from those series works of *Polar-vision*, we will integrate it to the community of LiDAR. A more ambitious idea is to introduce the concept of Riemannian geometry (Petersen, 2006) to regain the principals of 3d computer vision. We will nominate those series of works as *Riemannian-vision*.

Acknowledgements

References

Hartley, R., & Zisserman, A. (2003). Multiple view geometry in computer vision. Cambridge University Press.

Zhang, Z. (2000) A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, *22*(11), 1330-1334.

Schonberger, J. L., & Frahm, J. M. (2016) Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4104-4113).

Goesele, M., Snavely, N., Curless, B., Hoppe, H., & Seitz, S. M. (2007, October). Multi-view stereo for community photo collections. In *2007 IEEE 11th International Conference on Computer Vision* (pp. 1-8). IEEE.

Ryan, P. J. (1986). Euclidean and non-Euclidean geometry: an analytic approach. Cambridge university press.

Rosenfeld, A. (1988). Computer vision: basic principles. Proceedings of the IEEE, 76(8), 863-868.

Triggs, B., McLauchlan, P. F., Hartley, R. I., & Fitzgibbon, A. W. (2000). Bundle adjustment—a modern synthesis. In Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings (pp. 298-372). Springer Berlin Heidelberg.

D. R. Li. Basic photogrammetry [M]. Beijing: Surveying and Mapping Press, 1995. (In Chinese)

E. M. Mikhail, J. S. Bethel, J. C. (2001) Mcglone. Introduction to modern photogrammetry [M]. New York: Chichester Wiley.

Brown, D. (1996). Decentering distortion of lenses. Photogrammetric engineering, 32(3), 444-462.

A. E. Conrady. (1919). Decentred Lens-Systems, Monthly Notices of the Royal Astronomical Society, 79(5), 384–390.

Kannala, J., & Brandt, S. S. (2006). A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. IEEE transactions on pattern analysis and machine intelligence, *28*(8), 1335-1340.

Yang, Y., Chen, J. X., & Beheshti, M. (2005). Nonlinear perspective projections and magic lenses: 3D view deformation. IEEE Computer Graphics and Applications, *25*(1), 76-84.

Chang, C. H., Hu, M. C., Cheng, W. H., & Chuang, Y. Y. (2013). Rectangling stereographic projection for wide-angle image visualization. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2824-2831).

Hughes, C., McFeely, R., Denny, P., Glavin, M., & Jones, E. (2010). Equidistant (fθ) fish-eye perspective with application in distortion centre estimation. Image and Vision Computing, 28(3), 538-551.

Eichenseer, A., Bätz, M., Seiler, J., & Kaup, A. (2015, September). A hybrid motion estimation technique for fisheye video sequences based on equisolid re-projection. In 2015 IEEE International Conference on Image Processing (ICIP) (pp. 3565-3569). IEEE.

Zhang, X., Yu, F. X., Guo, R., Kumar, S., Wang, S., & Chang, S. F. (2015). Fast orthogonal projection based on kronecker product. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2929-2937).

Lee, S. H., & Civera, J. (2019). Closed-form optimal two-view triangulation based on angular errors. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 2681-2689).

Z. Z. Wang. Photogrammetric Principle [M]. Wuhan: Wuhan University Press, 2007. (In Chinese)

X. L. Liu. Principle and Technology of Ome-array Digital Aerophotogrammetry [M]. Zhengzhou: Henan Science and Technology Press, 2013. (In Chinese)

Z. X. Zhang. 30 Years of Digital Photogrammetry Research [M]. Wuhan: Wuhan University Press, 2007. (In Chinese)

R. X. Wang. Satellite Photogrammetric Principle for Three-Line-Array CCD Imagery [M]. Beijing: Surving and Mapping Press, 2016. (In Chinese)

Mur-Artal, R., Montiel, J. M. M., & Tardos, J. D. (2015). ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE transactions on robotics, 31(5), 1147-1163.

Mur-Artal, R., & Tardós, J. D. (2017). Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. IEEE transactions on robotics, 33(5), 1255-1262.

Cole, D. M., & Newman, P. M. (2006, May). Using laser range data for 3D SLAM in outdoor environments. In Proceedings 2006 IEEE International Conference on Robotics and Automation (ICRA) (pp. 1556-1563). IEEE.

Deans, M. C. (2005). Bearings-only* localization and mapping. Carnegie Mellon University.

Chiuso, A., Favaro, P., Hailin Jin, & Soatto, S. (2002). Structure from motion causally integrated over time. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(4), 523–535.

Bailey, T. (2003). Constrained initialisation for bearing-only SLAM. In 2003 IEEE International Conference on Robotics and Automation (ICRA). (Vol. 2, pp. 1966-1971). IEEE.

Davison. (2003, October). Real-time simultaneous localisation and mapping with a single camera. In Proceedings Ninth IEEE International Conference on Computer Vision (pp. 1403-1410). IEEE.

Sola, J., Monin, A., Devy, M., & Lemaire, T. (2005, August). Undelayed initialization in bearing only SLAM. In 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (pp. 2499-2504). IEEE.

Sola, J., Monin, A., Devy, M., & Vidal-Calleja, T. (2008). Fusing Monocular Information in Multicamera SLAM. IEEE Transactions on Robotics, 958–968.

Terejanu, G. A. (2008). Extended kalman filter tutorial. University at Buffalo, 27.

Civera, J., Davison, A. J., & Montiel, J. (2008). Inverse depth parametrization for monocular SLAM. IEEE Transactions on Robotics, 932–945.

Sola, J., Vidal-Calleja, T., Civera, J., & Montiel, J. M. M. (2012). Impact of landmark parametrization on monocular EKF-SLAM with points and lines. International journal of computer vision, 97, 339-368.

Eade, E., & Drummond, T. (2006, June). Scalable monocular SLAM. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06) (Vol. 1, pp. 469-476). IEEE.

Konolige, K., & Agrawal, M. (2008). FrameSLAM: From bundle adjustment to real-time visual mapping. IEEE Transactions on Robotics, 24(5), 1066-1077.

Zhao, L., Huang, S., Sun, Y., Yan, L., & Dissanayake, G. (2015). Parallaxba: bundle adjustment using parallax angle feature parametrization. The International Journal of Robotics Research, 34(4-5), 493-516.

Lourakis, M. I., & Argyros, A. A. (2009). SBA: A software package for generic sparse bundle adjustment. ACM Transactions on Mathematical Software (TOMS), 36(1), 1-30.

Konolige, K., & Garage, W. (2010, September). Sparse Sparse Bundle Adjustment. In BMVC (Vol. 10, pp. 102-1).

Z. K. Zuo, L. Yan, Y. B. Sun., et al. Applicability study on parallax bundle adjustment in 3D-photography [J]. Acta Scientiarum Naturalium Universitatis Pekinensis, 2023, 59(03): 445-455. (In Chinese)

Y. B. Sun. Study on convergence and convergence speed of polar coordinate bundle adjustment model [D]. Beijing: Peking University, 2015. (In Chinese)

IEEE 1937.11-2023, IEEE Standard for Technical Requirements of Polar Coordinate Photogrammetry Based on Unmanned Aircraft Systems[S].

Z. K. Zuo. Polar-coordinated vision theory & stereo pipeline for satellite [D]. Beijing: Peking University, 2023. (In Chinese)

Foley, J. D. (1996). Computer graphics: principles and practice (Vol. 12110). Addison-Wesley Professional.

Euler, L. (1765). Theoria motus corporum solidorum seu rigidorum (etc.) (Cum tabulis aeneis.). AF Röse.

Weisstein, E. W. (2009). Euler angles. https://mathworld. wolfram. com/.

Brezov, D. S., Mladenova, C. D., & Mladenov, I. M. (2013, December). New perspective on the gimbal lock problem. In AIP Conference Proceedings (Vol. 1570, No. 1, pp. 367-374). American Institute of Physics.

Caccavale, F., Natale, C., Siciliano, B., & Villani, L. (1999). Six-dof impedance control based on angle/axis representations. IEEE Transactions on Robotics and Automation, 15(2), 289-300.

Rodrigues, O. (1840). Des lois géométriques qui régissent les déplacements d'un système solide dans l'espace, et de la variation des coordonnées provenant de ces déplacements considérés indépendamment des causes qui peuvent les produire. Journal de mathématiques pures et appliquées, 5, 380-440.

Gray, J. J. (1980). Olinde Rodrigues' paper of 1840 on transformation groups. Archive for History of Exact Sciences, 375-385.

Murray, R. M., Li, Z., & Sastry, S. S. (2017). A mathematical introduction to robotic manipulation. CRC press.

Moulon, P., Monasse, P., Perrot, R., & Marlet, R. (2017). Openmvg: Open multiple view geometry. In Reproducible Research in Pattern Recognition: First International Workshop, RRPR 2016, Cancún, Mexico, December 4, 2016, Revised Selected Papers 1 (pp. 60-74). Springer International Publishing.

Craig, J. J. (2005). Introduction to Robotics: Mechanics and Control, Third.

Hamilton, W. R. (1840). On a new species of imaginary quantities, connected with the theory of quaternions. Proceedings of the Royal Irish Academy (1836-1869), 2, 424-434.

Cardano, G., & Spon, C. (1968). Ars magna (1545). Opera Omnia, 4, 221-302.

Lie, S. (1880). Über die Integration durch bestimmte Integrale von einer Classe linearer partieller Differentialgleichungen. Cammermeyer.

Lie, S. (1888). Classification und Integration von gewöhnlichen Differentialgleichungen zwischen xy, die eine Gruppe von Transformationen gestatten: Die nachstehende Arbeit erschien zum ersten Male im Frühling 1883 im norwegischen Archiv. Mathematische Annalen, 32, 213-281.

Sola, J., Deray, J., & Atchuthan, D. (2018). A micro Lie theory for state estimation in robotics. arXiv preprint arXiv:1812.01537.

Cauchy, A. L. (1815). Mémoire sur les fonctions qui ne peuvent obtenir que deux valeurs égales et de signes contraires par suite des transpositions opérées entre les variables qu'elles renferment. Journal de l'Ecole polytechnique, 10(17), 29-112.

Weisstein, E. W. (2003). Rotation matrix. https://mathworld. wolfram. com/.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., & Ng, R. (2021). Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM, 65(1), 99-106.

Kerbl, B., Kopanas, G., Leimkühler, T., & Drettakis, G. (2023). 3D Gaussian Splatting for Real-Time Radiance Field Rendering. ACM Transactions on Graphics, 42(4).

Chen, G., & Wang, W. (2024). A Survey on 3D Gaussian Splatting. arXiv preprint arXiv:2401.03890.

Petersen, P. (2006). Riemannian geometry (Vol. 171, pp. xvi+-401). New York: Springer.