Omnidirectional Cameras

Luiz Velho
IMPA

Outlook

- Practice
  - Wide-Angle Image Capture
  - 360 Cameras Rigs

- Theory
  - Geometry of Omnidirectional Cameras
  - Camera Models

Practice

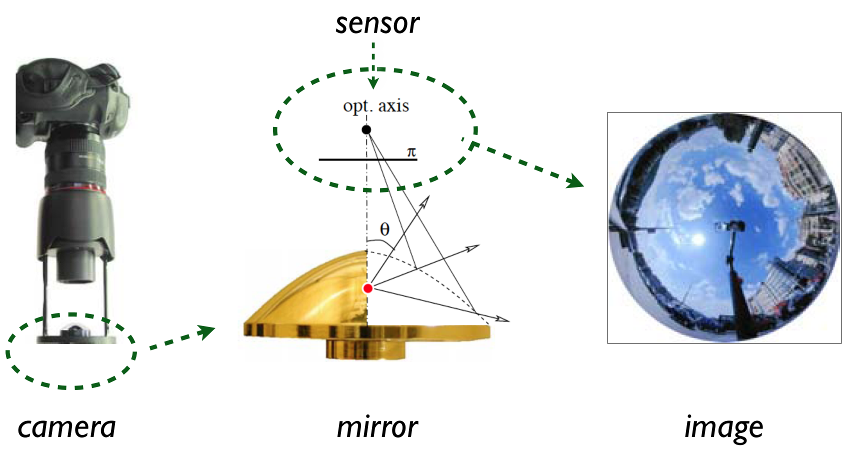
Wide-Angle Image Capture

Wide-Angle Image Capture

- Omnidirectional Cameras
  - Catadioptric
  - Dioptric

Catadioptric Cameras

- Mirror-Based (parabolic or hyperbolic)



Dioptric Cameras

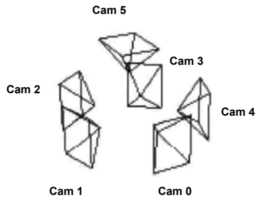
- Fish Eye Lenses

lens *camera* *image*

Omnidirectional Rigs

Multi-Camera Systems

- Point Grey's Ladybug (6 Perspective Cameras)



360 Panoramic

(2D Light Field)

---

360 Camera Rigs

360° Cameras

- Research Prototypes
- Professional Cameras

Research Prototypes

- Google Jump

2015

- Facebook Surround

2016

3D Stereo Cameras

(3D Light Field)

Professional Cameras

- Insta 360 Pro 2

  2018

3D Stereo

(3D Light Field)

- Lytro Imerge

  (2017 - 2018)

Light Field

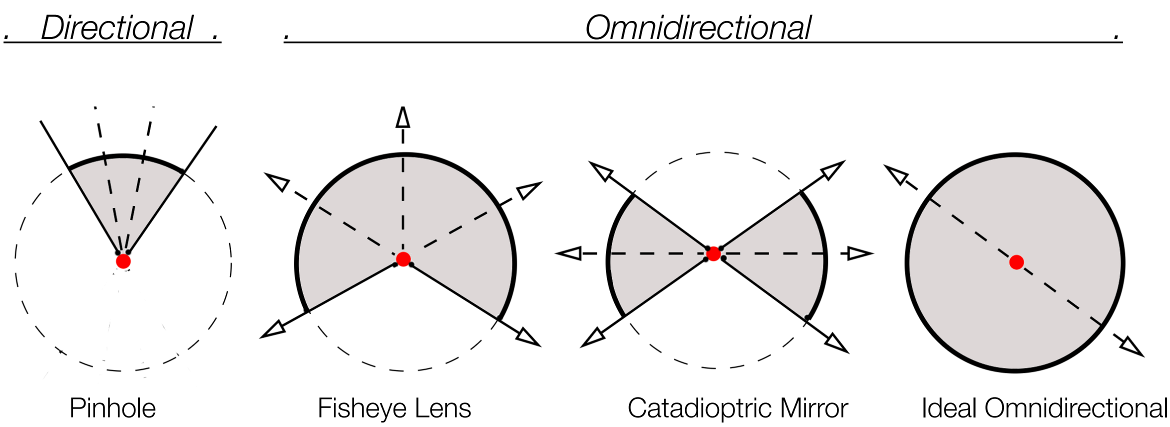
(4D Light Field)

Theory

Geometry of Omnidirectional Cameras

Based on Micusik, 2004

Camera Types

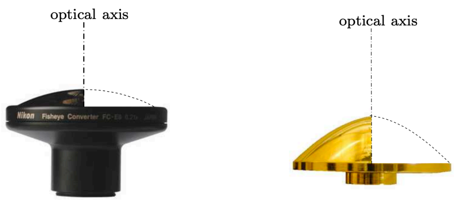


- Field of View

Image Formation

Key Assumptions

- Lenses & Mirrors:
  - Symmetric wrt. Optical Axis

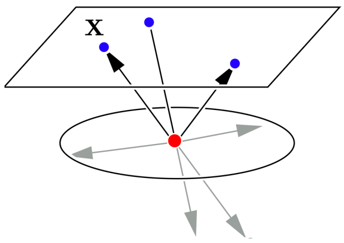


- Sensor Plane:
  - Perpendicular to Axis of Symmetry

Central Camera Models

- $P \in \mathbb{R}^{3\times4}$ (projection matrix), $X \in \mathbb{R}^4 \backslash \{0\}$ (scene point)

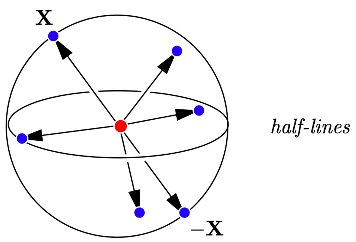
<u>*Perspective Model*</u>



$$\exists \alpha \neq 0: \ \alpha \mathbf{x} = \mathtt{P}\,\mathbf{X},$$

$\mathbf{x} \in \mathbb{R}^3 \setminus \{0\}$ image <u>point</u>.

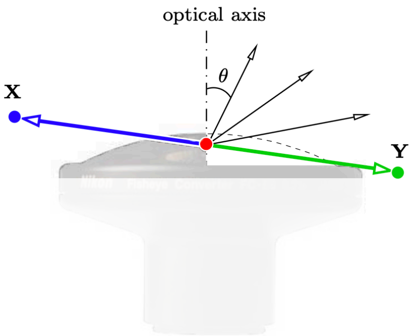
<u>*Spherical Model*</u>

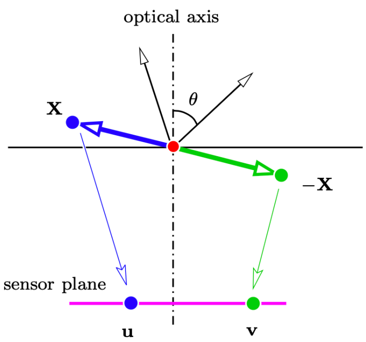


$$\exists \alpha > 0: \ \alpha \mathbf{q} = \mathtt{P}\,\mathbf{X},$$

$\mathbf{q} \in \mathbb{R}^3 \backslash \{0\}$ 3D <u>vector</u>

Omnidirectional Projection

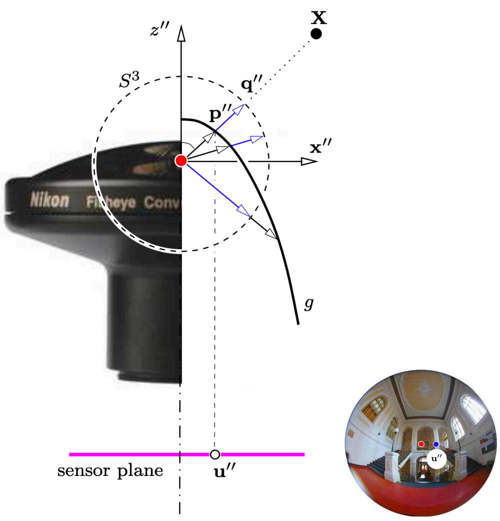




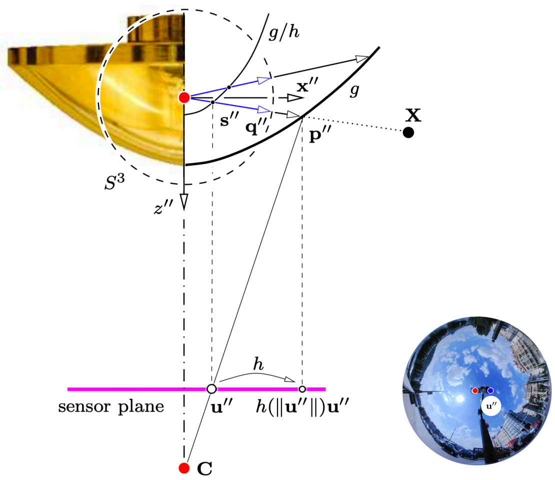
- Omnidirectional projection of two scene points X and Y=−X lying on opposite half-lines (map as two different image points **u** and **v** to a sensor plane)

Mapping to Sensor

- Projection of scene point $\mathbf{X}$ into a sensor plane point $\mathbf{u}''$



Fisheye Lens



Hyperbolic Mirror

Projection Model

- The projection of scene <u>point</u> **X** on the unit sphere around projection center **C**

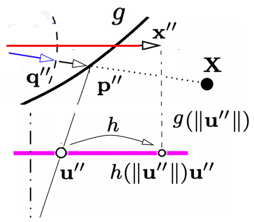
$$X \mapsto \mathbf{q}'' \in S^3 = \{\mathbf{x} \in \mathbb{R}^3 : \|\mathbf{x}\| = 1\}.$$

- There is a <u>vector</u> $\mathbf{p}'' = (\mathbf{x}'', z'')^T$ with *same direction* as $\mathbf{q}''$, which maps to *sensor plane* $\mathbf{u}''$, s.t. $\mathbf{u}''$ is *collinear* with $\mathbf{x}''$, i.e.

$$\mathbf{p}'' = \begin{pmatrix} h(\|\mathbf{u}''\|, \mathbf{a}'')\,\mathbf{u}'' \\ g(\|\mathbf{u}''\|, \mathbf{a}'') \end{pmatrix},$$

← *radial term*

← *axial term*



where g, h are functions $\mathbb{R} \times \mathbb{R}^N \to \mathbb{R}$, which depend on the radius $\|\mathbf{u}''\|$ of the sensor

Projection Functions

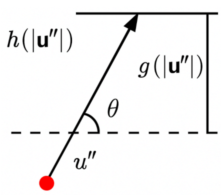
- Rotationally Symmetric

$$g, h(\|\mathbf{u}''\|) = g, h(\|\mathtt{R}\,\mathbf{u}''\|) \qquad \mathtt{R} \in \mathbb{R}^{2\times 2},$$ around the center of symmetry

- Depend on Camera Type
  - Lens Projection (Equisolid, Equiangular, Etc.)
  - Mirror Type (Parabolic, Hyperbolic, Elliptical)

- Function g
  - Physical Meaning (*shape of mirror*)

- Function h
  - Camera Projection (*i.e., Orthographic* $h = 1$)



General Projection

perspective projection: $\begin{pmatrix} 1\,\mathbf{u}'' \\ 1 \end{pmatrix}$

omnidirectional projection: $\begin{pmatrix} h(\|\mathbf{u}''\|)\,\mathbf{u}'' \\ g(\|\mathbf{u}''\|) \end{pmatrix}$

- Unified Model
  - Projection onto the unit sphere followed by a projection the sphere to a plane with a projection center on the perpendicular to the plane.

$$
\begin{aligned}
h(\|\mathbf{u}''\|) &= \frac{l(l+m) + \sqrt{\|\mathbf{u}''\|^2(1-l^2) + (l+m)^2}}{\|\mathbf{u}''\|^2 + (l+m)^2}\,, \\
g(\|\mathbf{u}''\|) &= \frac{l\|\mathbf{u}''\|^2 + (l+m)\sqrt{\|\mathbf{u}''\|^2(1-l^2) + (l+m)^2}}{\|\mathbf{u}''\|^2 + (l+m)^2}\,,
\end{aligned}
$$

where constants l, m, depending on the type of projection,

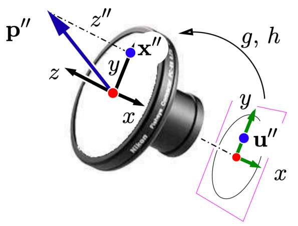
Digitization Process

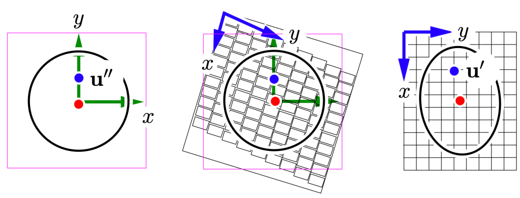
- 3 Steps:

  i) a central projection of the scene point $\mathbf{X}$ to the vector $\mathbf{p}''$,

  ii) a non-perspective optics or mirror reflection, described by the functions g and h, mapping $\mathbf{p}''$ to $\mathbf{u}''$, and

  iii) a digitization process transforming the sensor plane point $\mathbf{u}''$ to the digital image point $\mathbf{u}'$.





$$\mathbf{u}'' = \mathtt{A}'\mathbf{u}' + \mathbf{t}'\,,$$

Complete Image Formation

- Projection of a scene point $\mathbf{X}$ into the digital image point $\mathbf{u}'$

$$\frac{1}{\alpha''}\,\mathtt{P}''\,\mathbf{X} = \mathbf{p}'' = \begin{pmatrix} \mathbf{x}'' \\ z'' \end{pmatrix} = \begin{pmatrix} h(\|\mathbf{u}''\|)\mathbf{u}'' \\ g(\|\mathbf{u}''\|) \end{pmatrix} = \begin{pmatrix} h(\|\mathtt{A}'\mathbf{u}' + \mathbf{t}'\|)(\mathtt{A}'\mathbf{u}' + \mathbf{t}') \\ g(\|\mathtt{A}'\mathbf{u}' + \mathbf{t}'\|) \end{pmatrix},$$

so that the projection equation for omnidirectional cameras is

$$\exists\, \alpha'' > 0: \quad \alpha'' \begin{pmatrix} h(\|\mathtt{A}'\mathbf{u}' + \mathbf{t}'\|)(\mathtt{A}'\mathbf{u}' + \mathbf{t}') \\ g(\|\mathtt{A}'\mathbf{u}' + \mathbf{t}'\|) \end{pmatrix} = \mathtt{P}''\,\mathbf{X}\,,$$

where $\mathtt{P}'' \in \mathbb{R}^{3\times 4}$ is a projection matrix, $\mathtt{A}' \in \mathbb{R}^{2\times 2}$, $\operatorname{rank}(\mathtt{A}') = 2$, and $\mathbf{t}' \in \mathbb{R}^2$ represent an affine transformation in the sensor plane and $\mathbf{u}' \in \mathbb{R}^2$ is a point in the digital image.

Camera Models

Camera Models

- **Narrow** (< 180 degrees FOV - normal cameras)

Take in points that are in pixels (*normalized image coordinates)*.
Normalized image coordinates represent a 3D pointing vector within the FOV.

- **Wide** (no limit on FOV - fisheye or mirror)

Wide camera models use *spherical coordinates* instead since they do not make the assumption that visible points always appear in front of the camera.

Models

- Pinhole
- Brown
- Universal Omni
- Kannala-Brandt

Pinhole Model

- Basic camera model
  - Does not model lens distortion.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & skew & cx \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \\ 1 \end{bmatrix}$$

where the 3x3 is the intrinsic camera matrix and is also known as *K*.

Brown Model

- Models lens distortion radial and tangential ("decentering") distortion.
  - It is appropriate for most lenses with a FOV less than 180 degrees.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \sim K \begin{bmatrix} x_d \\ y_d \\ 1 \end{bmatrix}$$

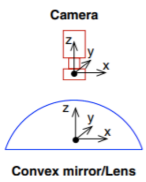
$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \sum_{i=0}^{i<rad} a_i r^{2i} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} 2t_1 x_n y_n + t_2(r^2 + 2x_n^2) \\ t_1(r^2 + 2y_n^2) + 2t_2 x_n y_n \end{bmatrix}$$

$r = \sqrt{x_n^2 + y_n^2}$ radial distortion coefficient

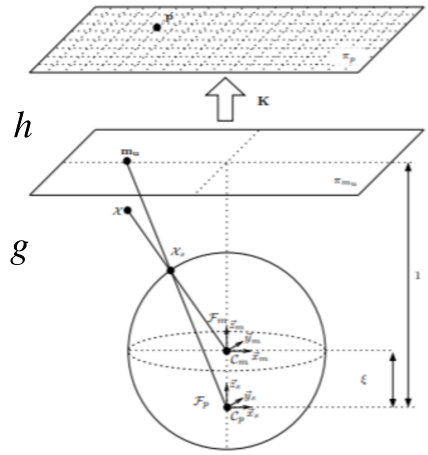
(t_1, t_2) tangential coefficients.

Universal Omni Model

- Adds a Mirror Offset to the Brown camera model
  - parabola, hyperbola, ellipse, and plane mirror
- Only Difference from the Brown model:
  - how it converts spherical into image coordinates:



$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = \begin{bmatrix} x_s/(z_s+\epsilon) \\ y_s/(z_s+\epsilon) \end{bmatrix}$$



- After this step it has identical equations to Brown

Kannala-Brandt Model

- Wide camera model that supports perspective, stereographic, equidistance, equisolid angle, and orthogonal projection models.

Radially Symmetric Model:

$$r(\theta) = k_1\theta + k_2\theta^3 + k_3\theta^5 + \cdots$$

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = r(\theta)\begin{bmatrix} \cos\phi \\ \sin\phi \end{bmatrix}$$

Radial Distortion Model:

$$\Delta_r(\theta,\phi) = (l_1\phi + l_2\phi^3 + l_3\phi^5)(i_1\cos\phi + i_2\sin\phi + i_3\cos 2\phi + i_4\sin 2\phi)$$

Tangential Distortion Model:

$$\Delta_t(\theta,\phi) = (m_1\phi + m_2\phi^3 + m_3\phi^5)(j_1\cos\phi + j_2\sin\phi + j_3\cos 2\phi + j_4\sin 2\phi)$$

Full Camera Model:

$$x_d = r(\theta)u_r(\phi) + \Delta_r(\theta,\phi)u_r(\phi) + \Delta_t(\theta,\phi)u_\phi(\phi)$$

where x_d are the distorted normalized image coordinates, $u_r(\phi)$ is a unit vector in radial direction, and $u_\phi(\phi)$ is a unit vector in tangential direction.