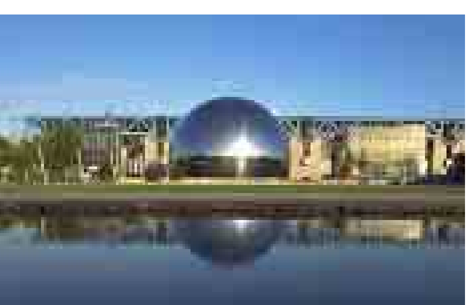
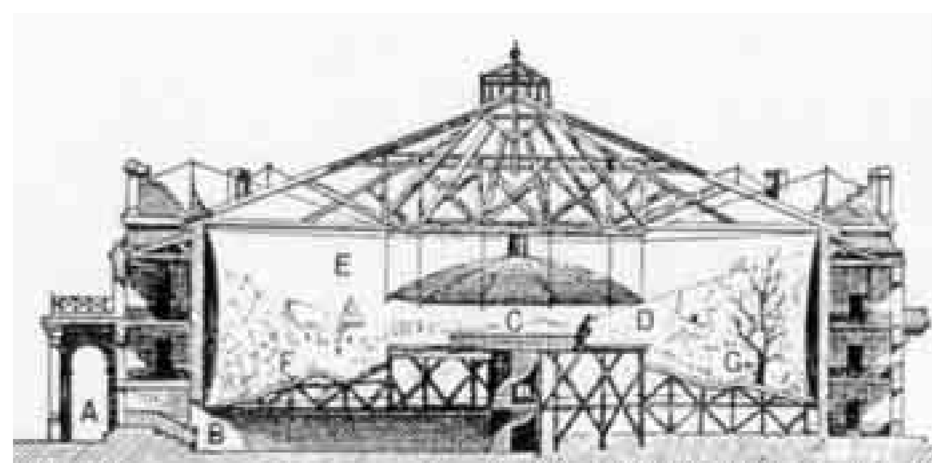
IEEE ITCC 2002. April 9th 2002
11:30 am, Las Vegas, USA.

High Resolution Full Spherical Videos

Frank Nielsen
Sony Computer Science Laboratories

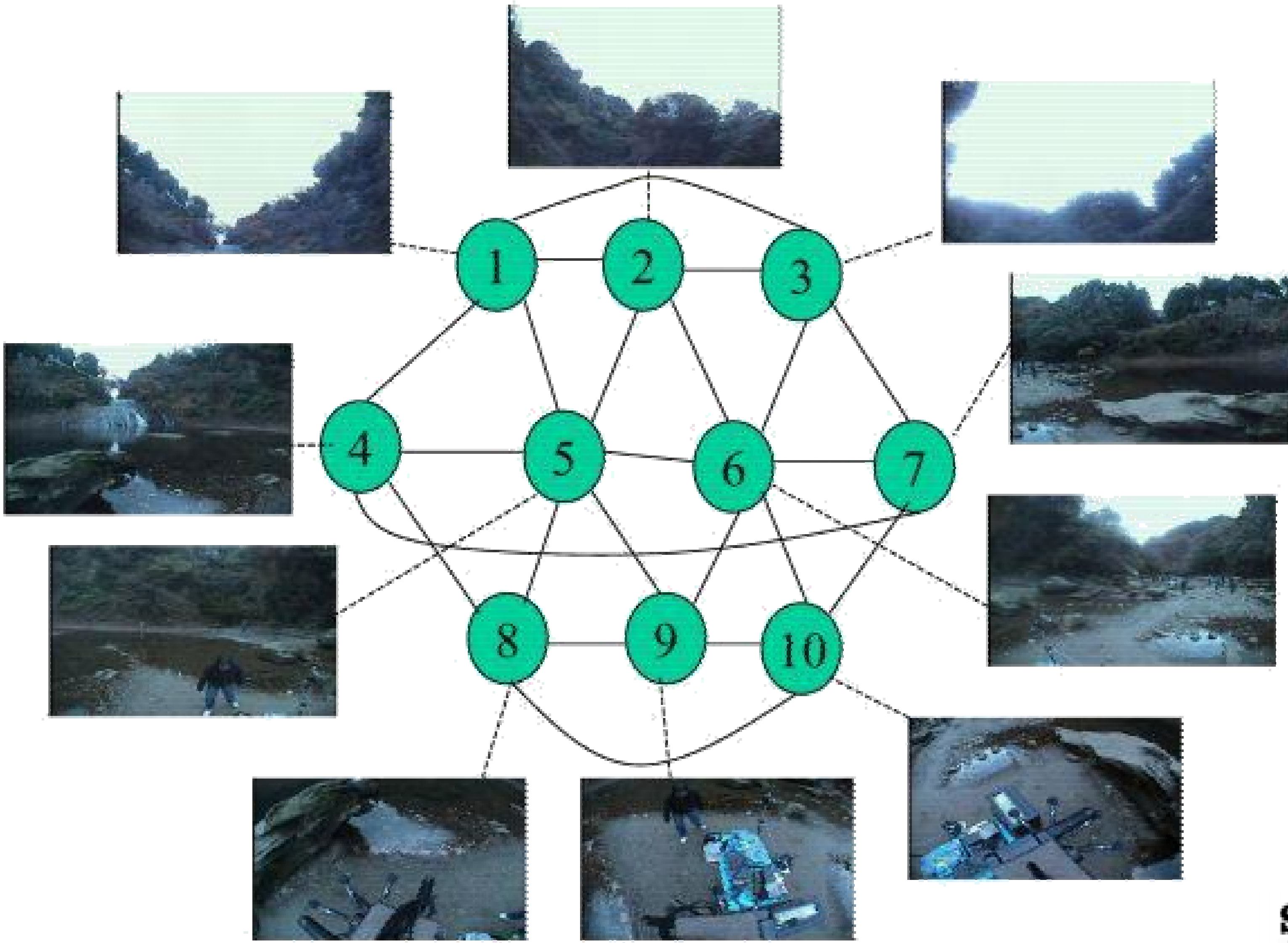




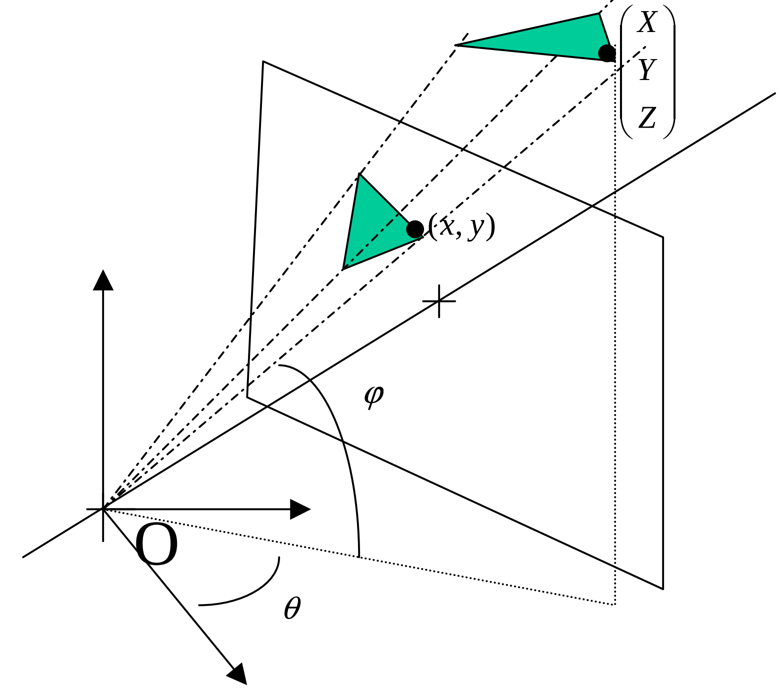
REGISTERING A 10-HEAD CAMERA UNIT

SONY CSL

IMAGE GRAPH FOR THE 10-HEAD CAMERAS



ABSTRACT FUNCTIONS



$$\text{RayToImage}(\theta, \varphi, C) = (x, y)$$

$$\text{ImageToRay}(x, y, C) = (\theta, \varphi)$$

(θ, φ) : Spherical Coordinates

(x, y) : Cartesian Coordinates

C : Camera model (pinhole, fisheye, Tsai's distortion, etc.)

- Do not resample original images → *preserve original quality*
- Combine several types of cameras → *flexibility*
- Allow field of view of units to be *bigger than 180 degrees*

EXAMPLE : RADIAL DISTORTIONS

Use Tsai generic model of camera (pincushion/barrel corrections)

ImageToRay

Compute ideal pinhole coordinates

$$\tilde{x} = x + (x - c_x)\kappa r^2$$

$$\tilde{y} = y + (y - c_y)\kappa r^2$$

$$\text{where } r^2 = (x - c_x)^2 + (y - c_y)^2$$

↓

Obtain 3d point on the unit sphere

$$P = K^{-1} \times \begin{pmatrix} \tilde{x} \\ \tilde{y} \\ 1 \end{pmatrix}, \text{ where } K = \begin{bmatrix} f_x & 0 & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$P_{\theta,\varphi} = \frac{P}{\|P\|} = \begin{pmatrix} X_{\theta,\varphi} \\ Y_{\theta,\varphi} \\ Z_{\theta,\varphi} \end{pmatrix}$$

↓

Get spherical coordinates

$$\varphi = \arcsin Y_{\theta,\varphi}$$

$$\theta = \arccos \frac{Z_{\theta,\varphi}}{\cos\varphi}$$

RayToImage

Obtain 3d point on the unit sphere

$$P_{\theta,\varphi} = \begin{pmatrix} \sin\theta\cos\varphi \\ \sin\varphi \\ \cos\theta\cos\varphi \end{pmatrix}$$

↓

Ideal cartesian pinhole coordinates

$$\begin{pmatrix} \tilde{x} & \tilde{y} \end{pmatrix} \cong \begin{pmatrix} \frac{x}{w} & \frac{y}{w} \end{pmatrix} = \begin{pmatrix} x & y & w \end{pmatrix} = K \times R \times P_{\theta,\varphi}$$

↓

Compute radial distortions

$$\tilde{r} = \sqrt{(\tilde{x} - c_x)^2 + (\tilde{y} - c_y)^2}$$

Solve $\kappa r^3 + r - \tilde{r} = 0$

Choose closest r' from r

↓

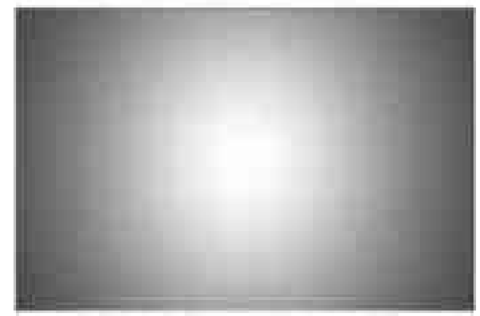
Obtain cartesian image coordinates

$$x = \frac{\tilde{x} + c_x \kappa r'^2}{1 + \kappa r'^2}, \quad y = \frac{\tilde{y} + c_y \kappa r'^2}{1 + \kappa r'^2}$$

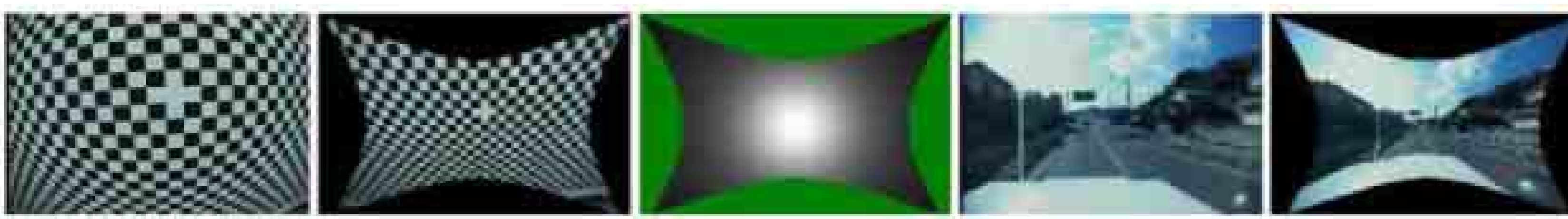
DENSITIES OF THE RAYS [RADIOMETRY]

1. In the image frame, fidelity of pixels (i.e. rays) are uneven!

Example of an ideal pinhole camera:



2. If model differs from ideal pinhole, divides by the stretching factor



Let $w(x, y)$ denotes the pixel fidelity

REGISTRATION (STITCHING)

- Initialize parameters by user GUI or calibration
- Numerically *optimize objective function* on selected set of parameters

<u>Numerical techniques in computer vision:</u>

- Levenberg-Marquadt process [all rayels]
- Bundle Adjustment [stellar features]

Many variables per camera:

- Absolute rotation: roll, pitch,yaw (3) [extrinsics]
- Lens center (2), distortions (2+3)
- Focals [fovs] (2), principal point (2)

Example of a 10-head camera: **140 parameters**

SONY CSL

LEVEMBERG MARQUADT

$$\mathbf{G} = - \sum_{(\theta,\phi)=m(x,y)|(x,y)\in\mathcal{R}} s_{\theta,\phi} \frac{\partial s_{\theta,\phi}}{\partial \mathbf{P}}$$

Given an env. map

Less defective

$$\mathbf{H}_{i,j} = - \sum_{(\theta,\phi)=m(x,y)|(x,y)\in\mathcal{R}} \frac{\partial^2 s_{\theta,\phi}}{\partial \mathbf{P}_i \partial \mathbf{P}_j}$$

$$\Delta\mathbf{P} = -(\mathbf{H} + \lambda\mathbf{I})^{-1}\mathbf{G}$$

REGISTRATION REFERENCES

- **Based on Newton or Levemberg-Marquadt procedure**

J. More. *The levenberg-marquardt algorithm*, implementation and theory.
In G. A. Watson, editor, Numerical Analysis, Lecture Notes in Mathematics 630. Springer-Verlag, 1977.

- **For per-pixel registration, use a pyramid level of images**

P. J. Burt and E. H. Adelson. *A multiresolution spline with application to image mosaics*.
ACM Transactions on Graphics, 2(4):217--236, Oct. 1983.
P. J. Burt and E. H. Adelson, *The Laplacian pyramid as a compact image code*,
IEEE Trans. on Communications, vol. COM-31, pp. 532-540,1983.

- **Phase correlation method (cartesian, affine only)**

C. D. Kuglin and D. C. Hines. *The **phase correlation** image alignment method*.
In IEEE 1975 Conference on Cybernetics and Society, pages 163--165, New York, September 1975.

- **Bundle Adjustment (model optimization+parameters)**

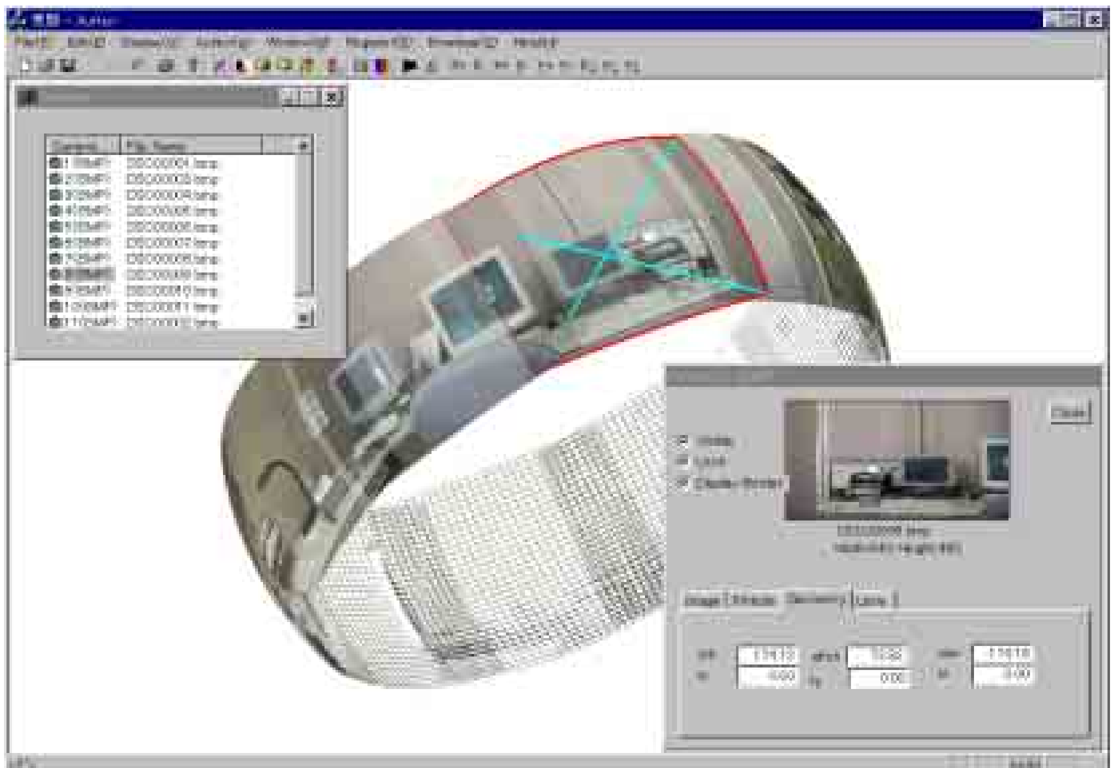
B. Triggs, P. McLauchlan, R. Hartley and A. Fitzgibbon
Bundle Adjustment -- A Modern Synthesis
Vision Algorithms: Theory and Practice, Springer Verlag, LNCS, pp. 298-375, 2000.

- **Optical flow**

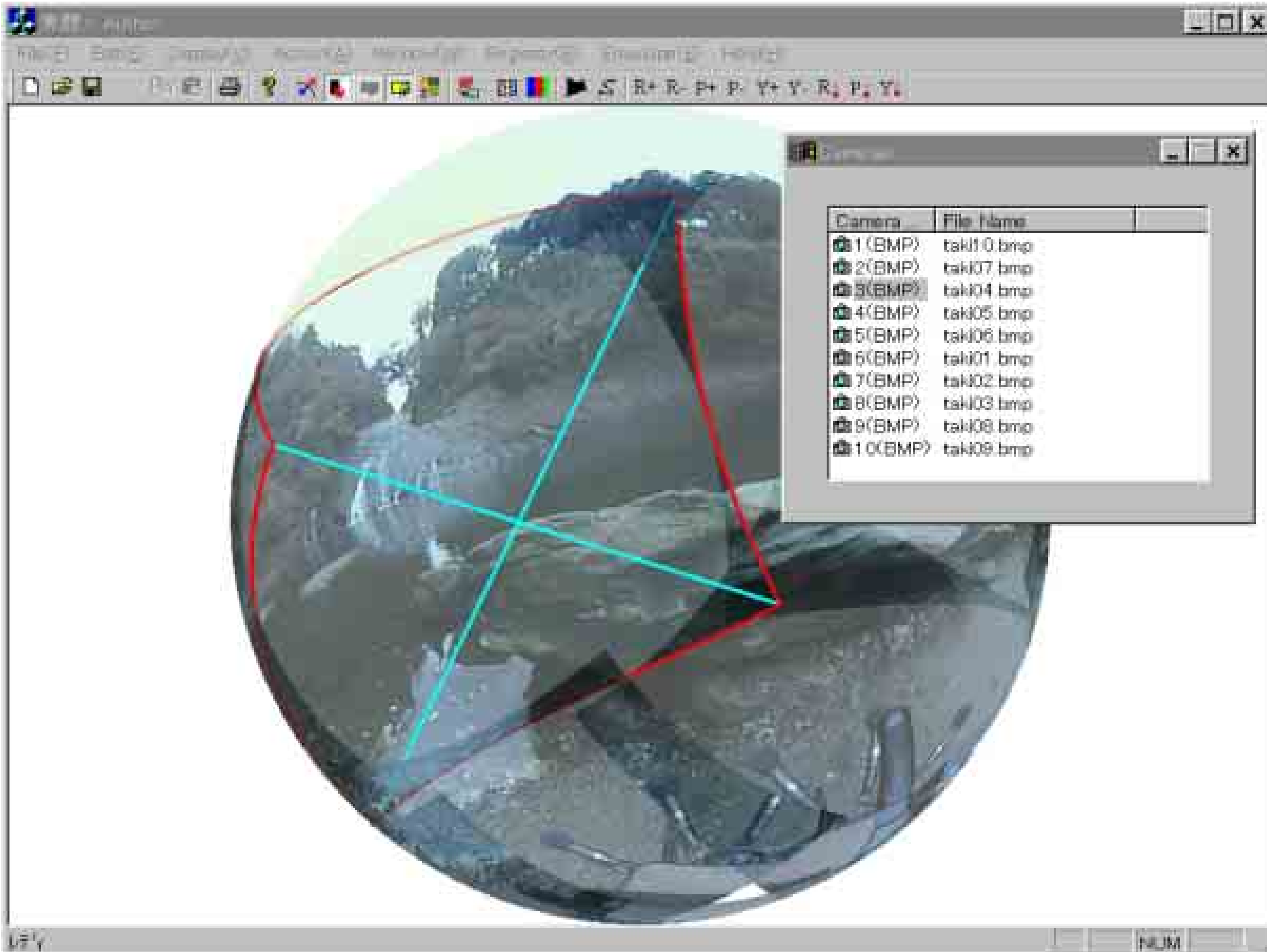
B K P Horn and B G Schunck. *Determining optical flow*. Artificial Intelligence, 17:185--203, 1981

SONY CSL

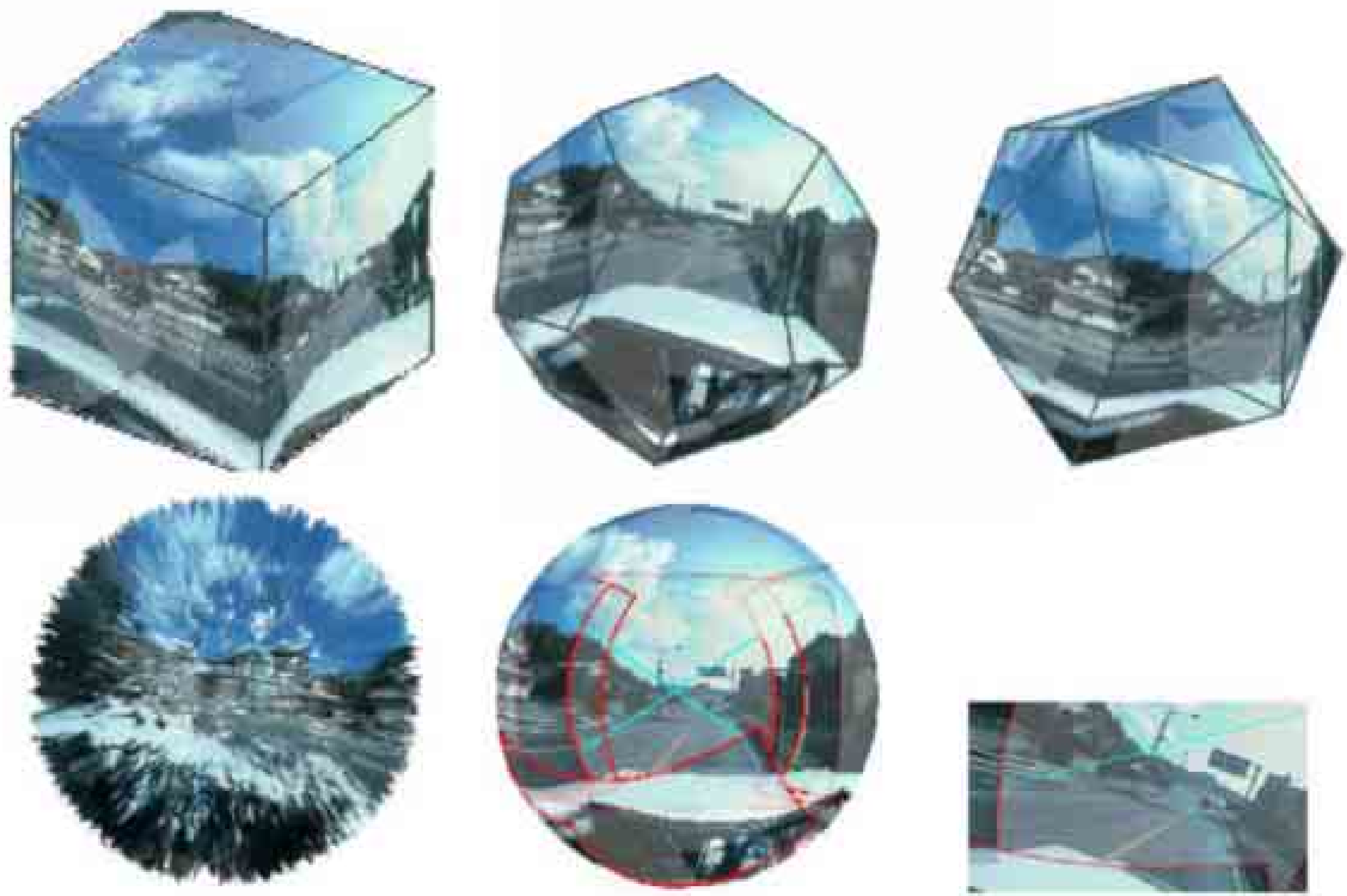
AUTHORING (I)



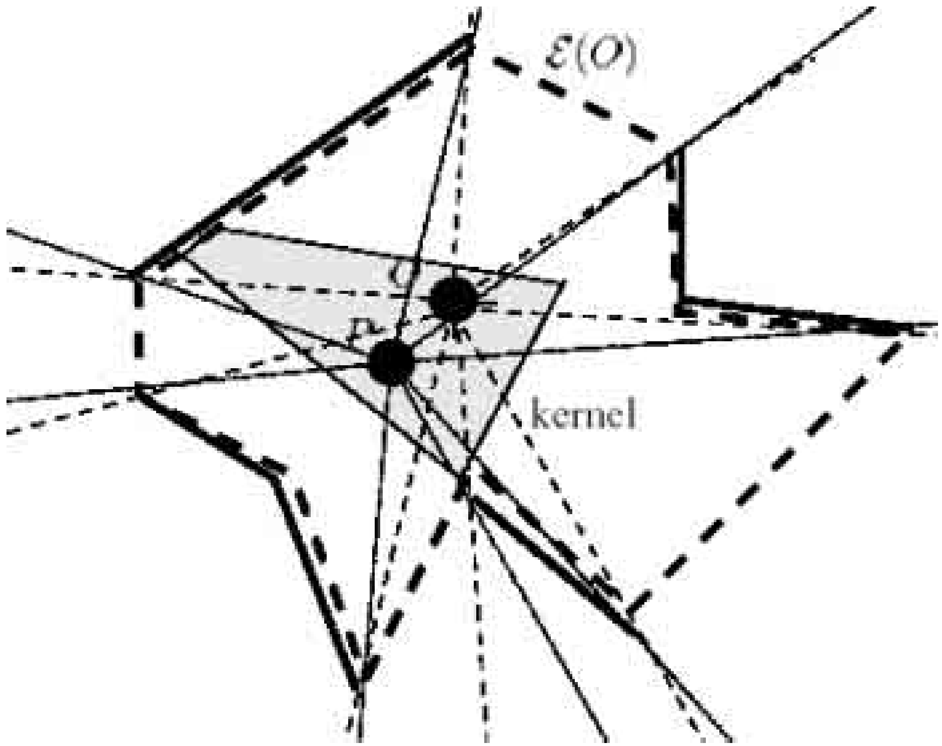
AUTHORING (II)



STITCHING ON ENVELOPES



ENVELOPES…

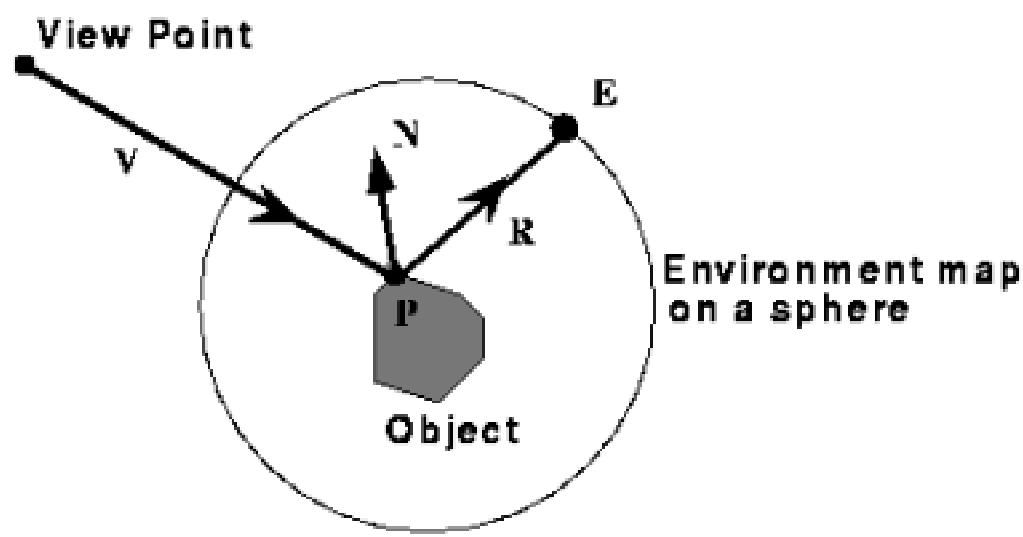


Allows the user to move in its visibility cell

SPHERICAL VIDEOS & ENVIRONMENT MAPS

- Spherical video captures all rays passing through a central point.

- Environment/light mappings map a set of rays to a computer graphics scene included into a ball. (➠ mipmapping)

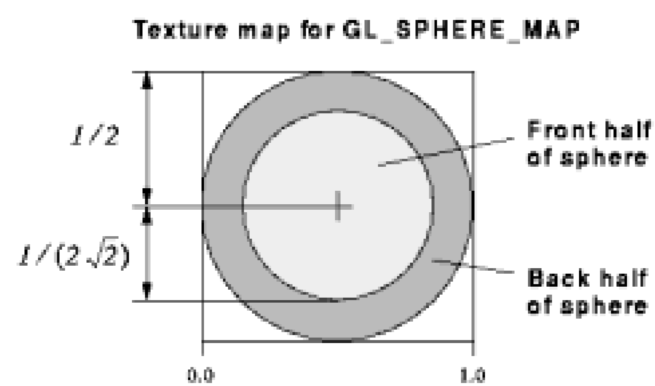


MAPPING: SPHERE MAP (GL_SPHERE_MAP)

Movie

Cons:

- Singularity at (0,0,-1)
- One d.o.f.
- Black areas of texture
- Non uniform density



MAPPING: LATITUDE LONGITUDE

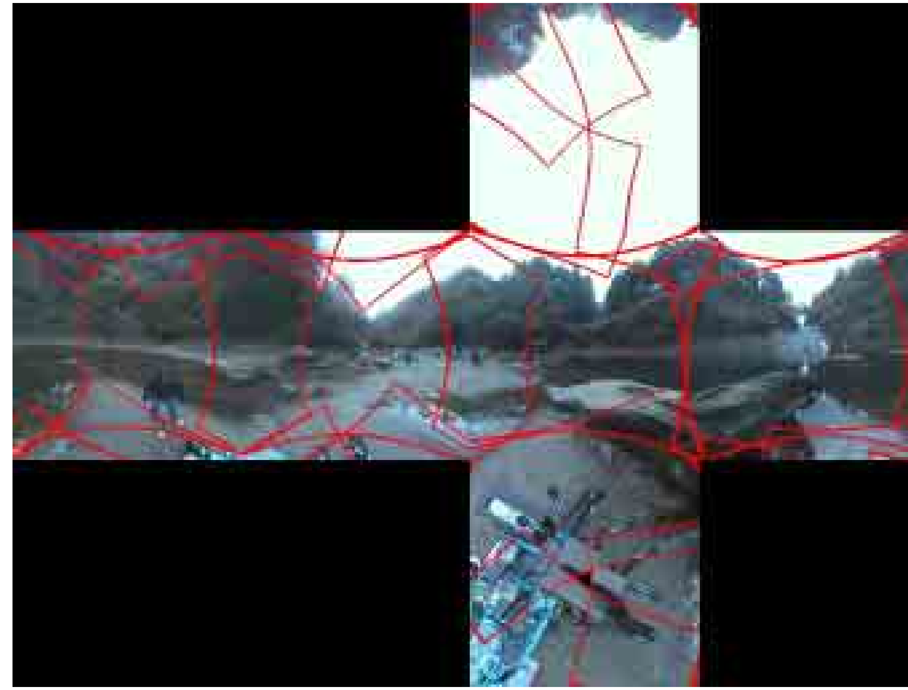
Blinn, J. F. and Newell, M. E.
Texture and reflection in computer generated images.
Communications of the ACM Vol. 19, No. 10 (1976), 542-547.

Also called equi-rectangular.   Cons: oversampling at the N.-S. poles.

MAPPING: CUBIC

- 6 faces (*hardware accelerated*)

Ex. Quicktime Cubic VR



•Using *software projective texture* (Open GL v1.2), can use our own packing of faces into a single map.

SONY CSL

MAPPING: DUAL PARABOLOID

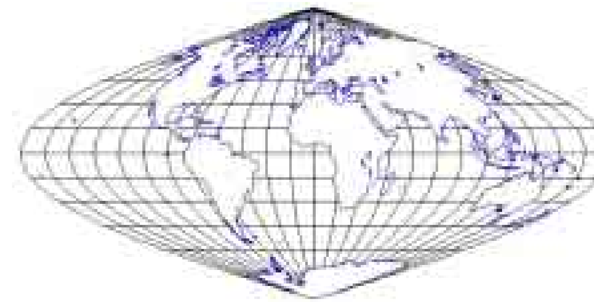
Wolfgang Heidrich, Hans-Peter Seidel, "View-independent Environment Maps,"
In *Proc. of the Eurographics/Siggraph Workshop on Graphics Hardware* 1998, pp. 39-45.

Software front and back maps (multi-texturing with alpha channel)

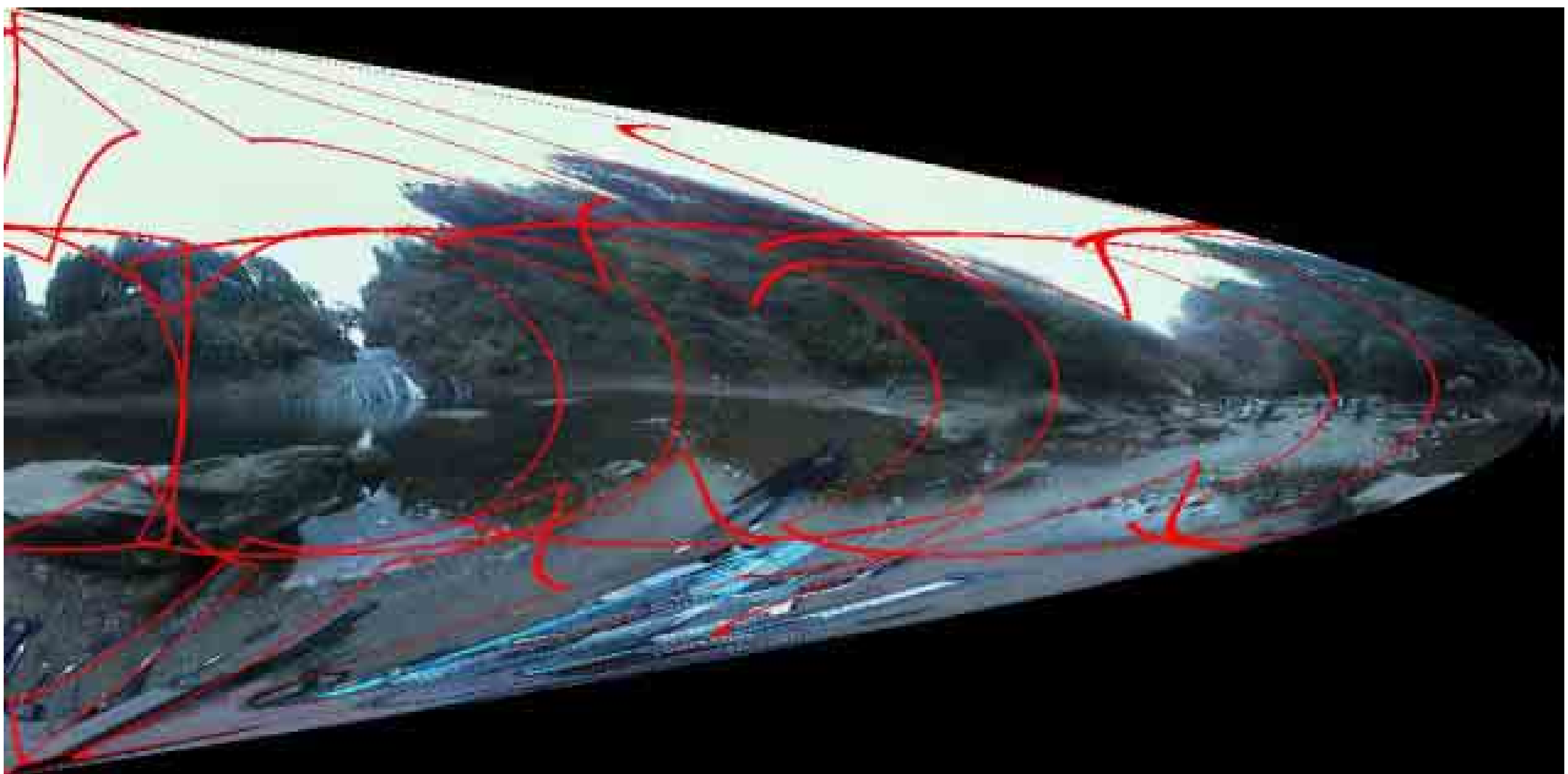
SONY CSL

MAPPING: COMPRESSED SPHERICAL

Inspired from Sanson's map in cartography

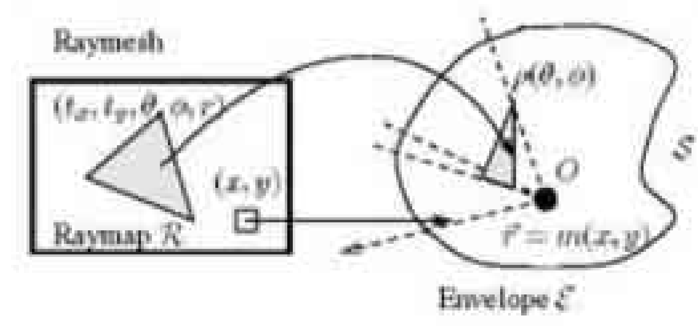


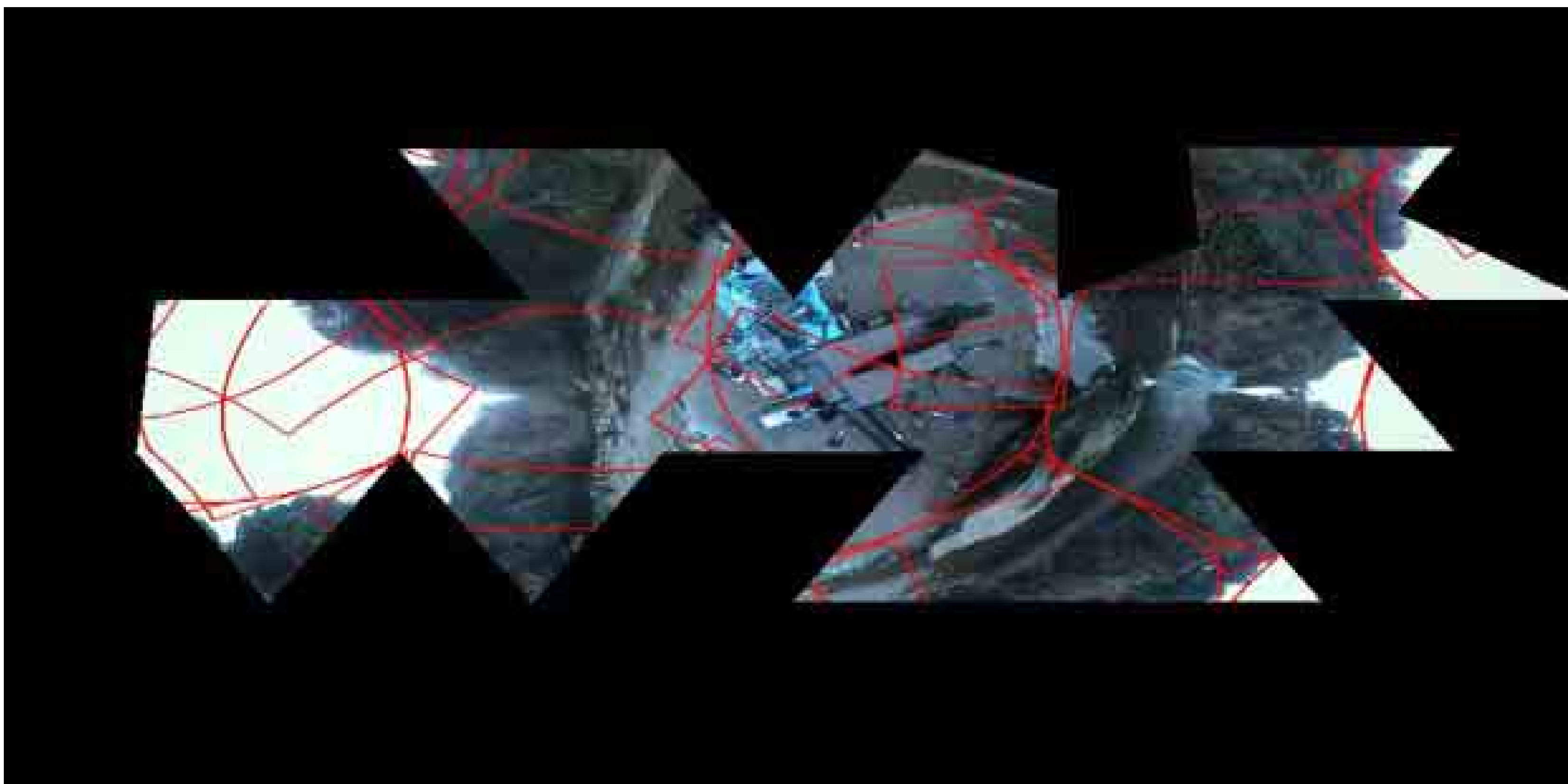
- Same quality as latitude-longitude (save 36%)
- Well suited to texture mapping
- Video compression based on MPEG (half border blocks).



EXAMPLE OF A RAYMESH: B. FULLER'S MAP

Also known as dymaxion®
Unfolding & face cutting of an icosahedron



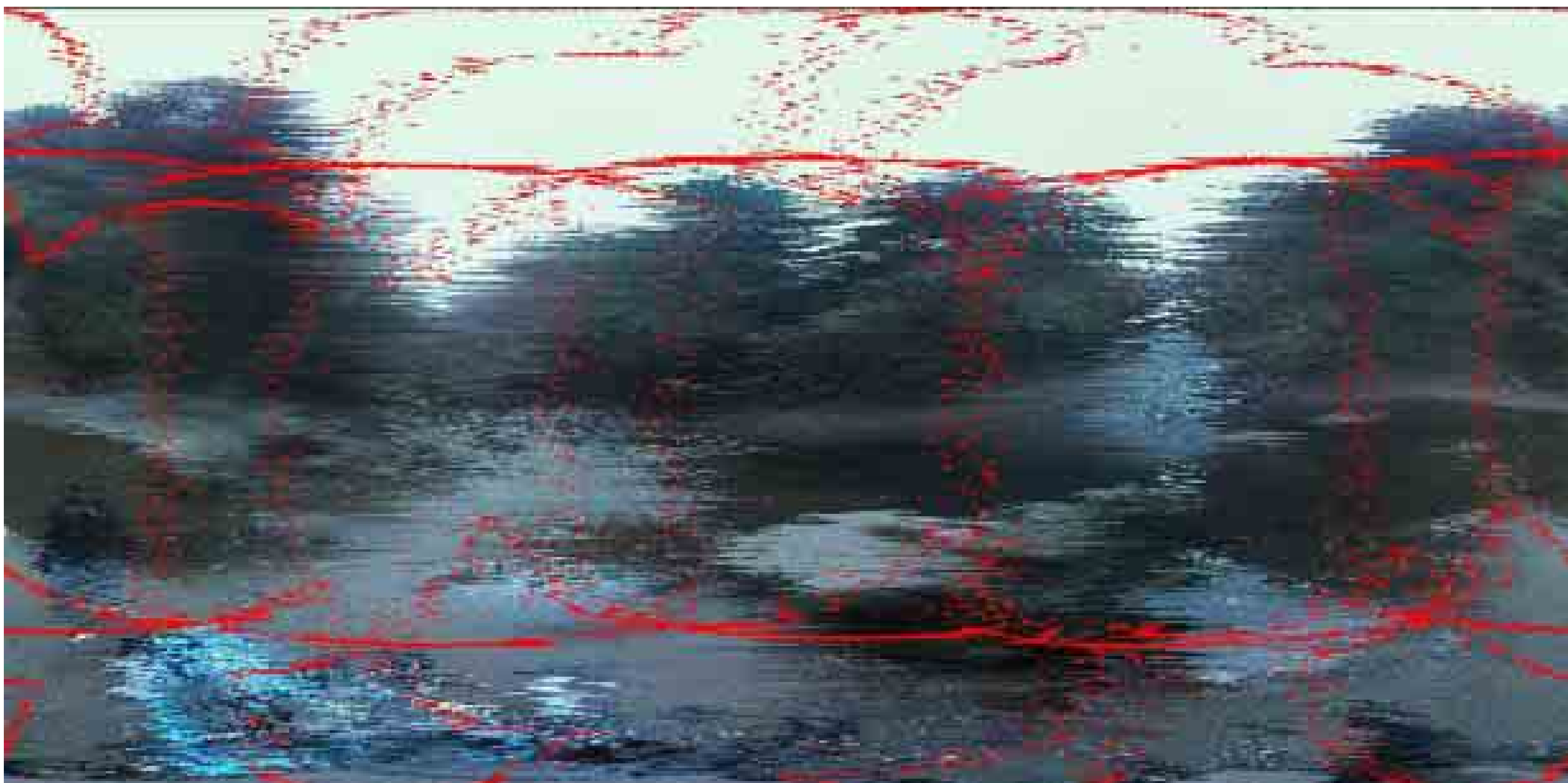


Unfolding movie

STRATIFIED RANDOM

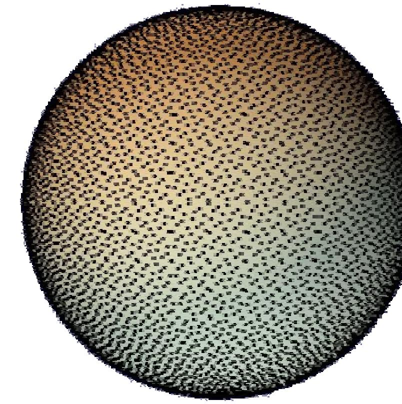
- Take sample points from annular slices of same slice width.
- Sort inside each slice lexicographically the thetas and store.
- Use a **random seed** to build the table correspondence (x,y,theta,phi).

Higher resolution than latitude-longitude but only suitable for per pixel element drawing

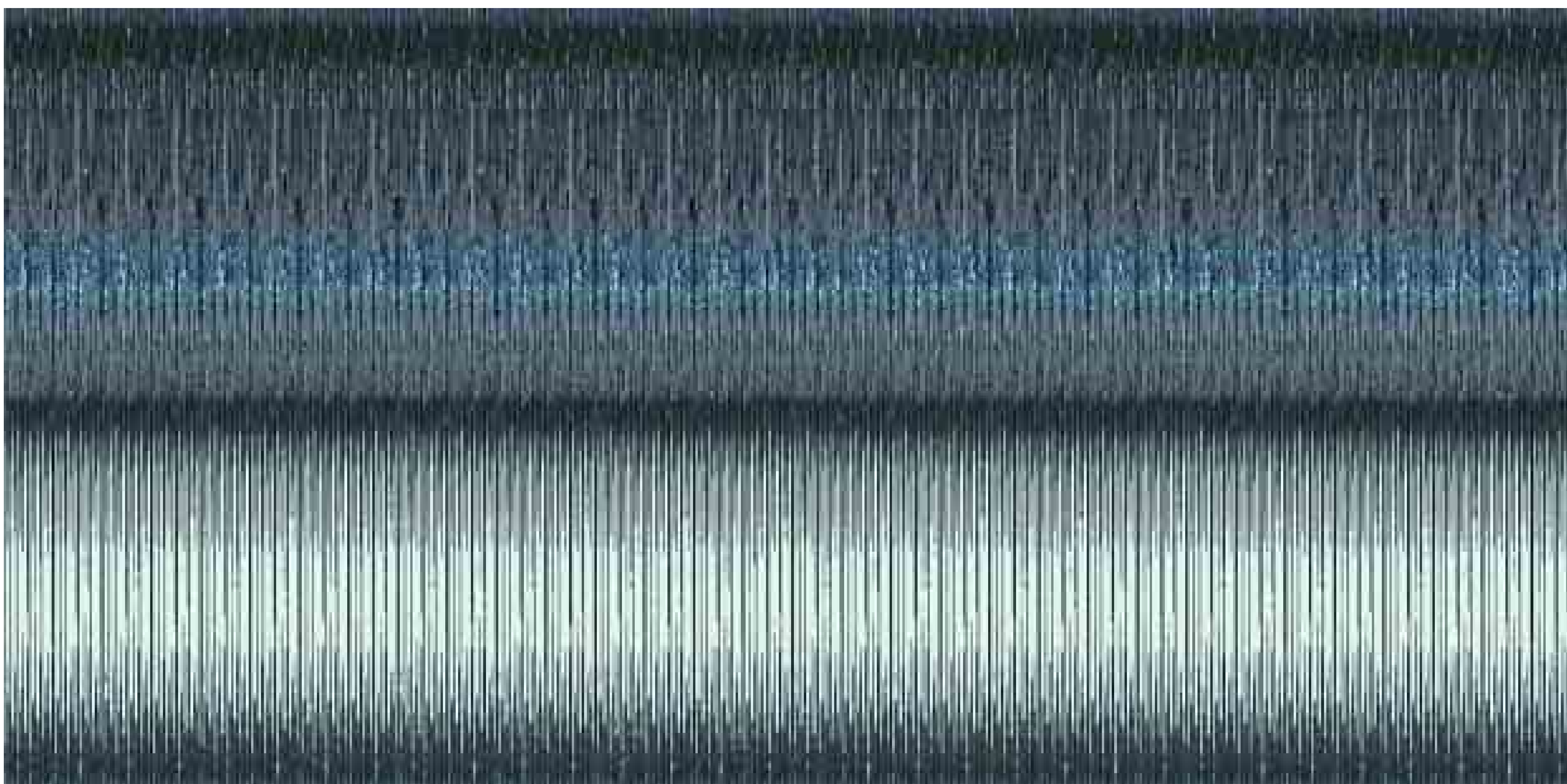


LOW DISCREPANCY SEQUENCES

(example of Hammersley sequences)



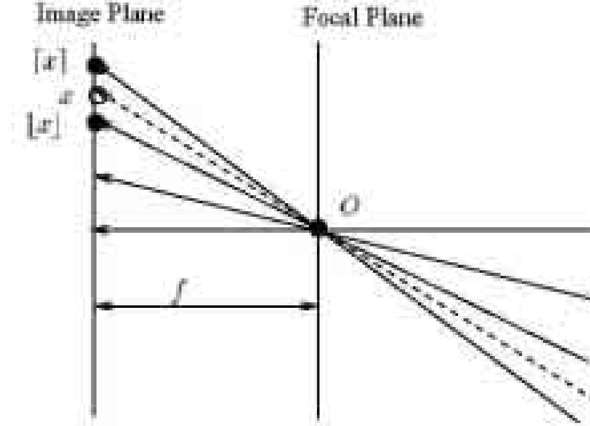
Proven *low discrepancy=good sampling* …
But pixel interpolation more complex



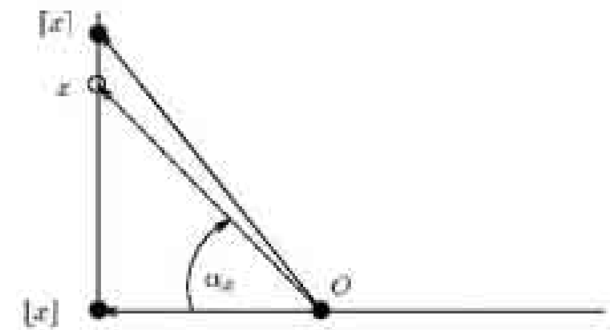
Point distribution demo

Format	*Advantages*	*Drawbacks*
Spherical	Hardware accelerated environment mappings.	For a given viewpoint.Singularities Resolution distribution ratio is bad
Cubical	Hardware accelerated environment mappings. Resolution distribution ratio is $3\sqrt{3}$	Need to give 6 different files
Dual paraboloid	Resolution distribution ratio is 4	Not hardware accelerated but using OpenGL can texture fast
Equi-rectangular	Easy to interpret. Widely used	Resolution distribution ratio is bad
Compressed spherical	Same info as of equi-rectangular but using less memory. Well suited to immersive video (based on sinusoidal maps in cartography)	Resolution distribution is bad
Ray mesh	**Flexible format. Well suited to immersive video** **Control the important area (eventually dynamic)**	**Users have to tell the system which dynamic mesh to use**
Low discrepancy	Uniform in every direction. Scalable progressively (interpolation hard)	Well suited to still imaging (video codec …)
Stratified random	Randomly uniform. Stratified per slide (interpolation easier)	Well suited to still imaging (video codec)

PIXEL/RAY INTERPOLATION



Interpolate in the rotation space
not in the usual cartesian 2d space



Important for:

- Per-pixel registration
- Output synthesis
- Etc. (always when using camera images?)

Linear pixel interpolation

$$r_x = x - \lfloor x \rfloor$$

$$a_x = (1 - r_x) a_{\lfloor x \rfloor} + r_x a_{\lceil x \rceil}$$

Linear ray interpolation

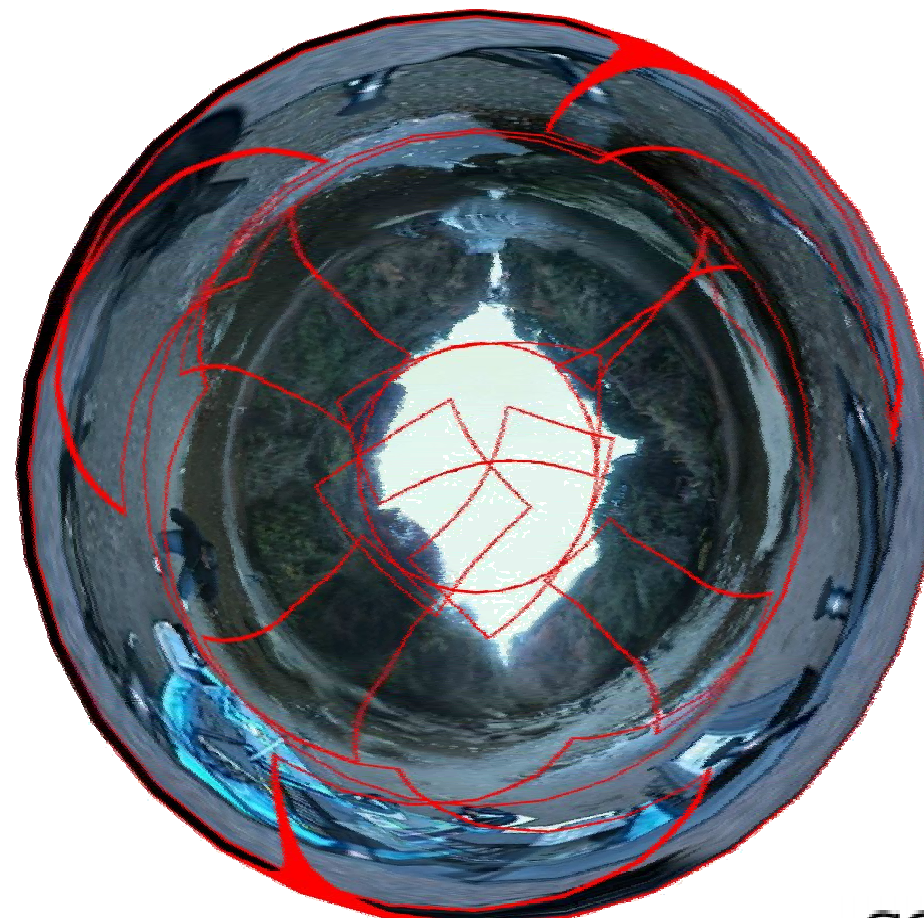
$$\alpha_x = \frac{\arctan \frac{x - p_x}{f_x} - \arctan \frac{\lfloor x \rfloor - p_x}{f_x}}{\arctan \frac{\lceil x \rceil - p_x}{f_x} - \arctan \frac{\lfloor x \rfloor - p_x}{f_x}}$$

$$a_x = (1 - \alpha_x) a_{\lfloor x \rfloor} + \alpha_x a_{\lceil x \rceil}$$

VIEWER

- Draw **per-pixel** or **per-triangle** primitives in either **2d/3d**.
- Can synthesize any projective camera view…

Example of an angular map
(fisheye 360 degree)

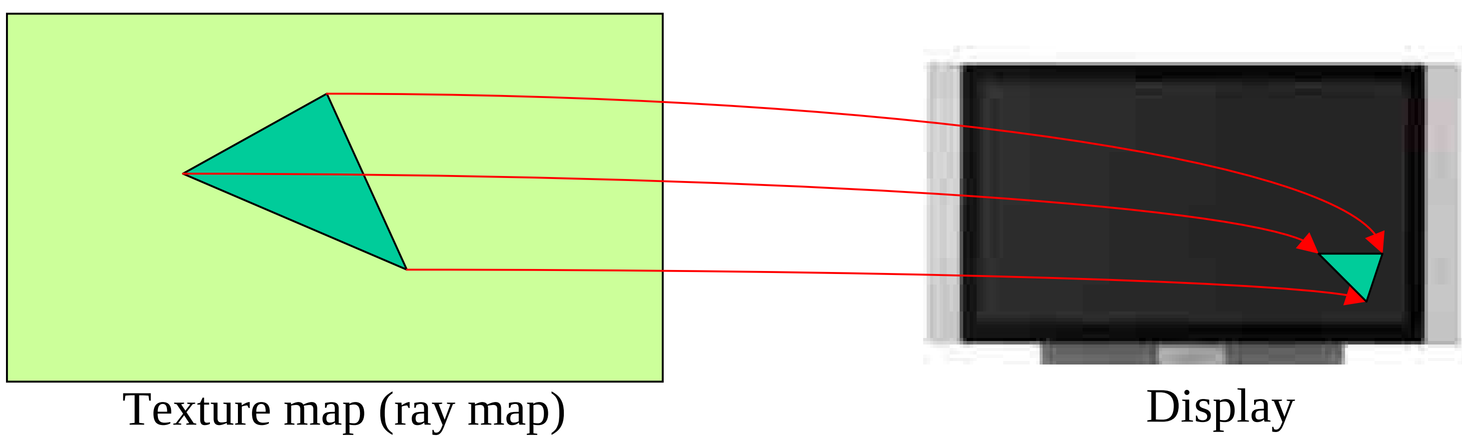


VIEWER

Example of drawing 2d per-triangle primitives.

$$(TX, TY) \xrightarrow{\text{Look up table}} (\theta, \varphi) \xrightarrow{\text{Select view}} (\theta', \varphi') \xrightarrow{\text{RayToImage}} (x, y)$$

(TX, TY): Texture coordinates



Hardware filtering operations

APPLICATIONS

Technology scenarios:

- User's own media experience
(Convergence of computer video, graphics and vision)

- Game skyboxes

- Video-based light rendering

- CG / Movie reflection mappings

- Net meetings, etc.

FINAL REMARKS

Spherical Movie Demo　　Spherical Movie CG

SPATIAL MEDIA PROJECT HOME PAGE:

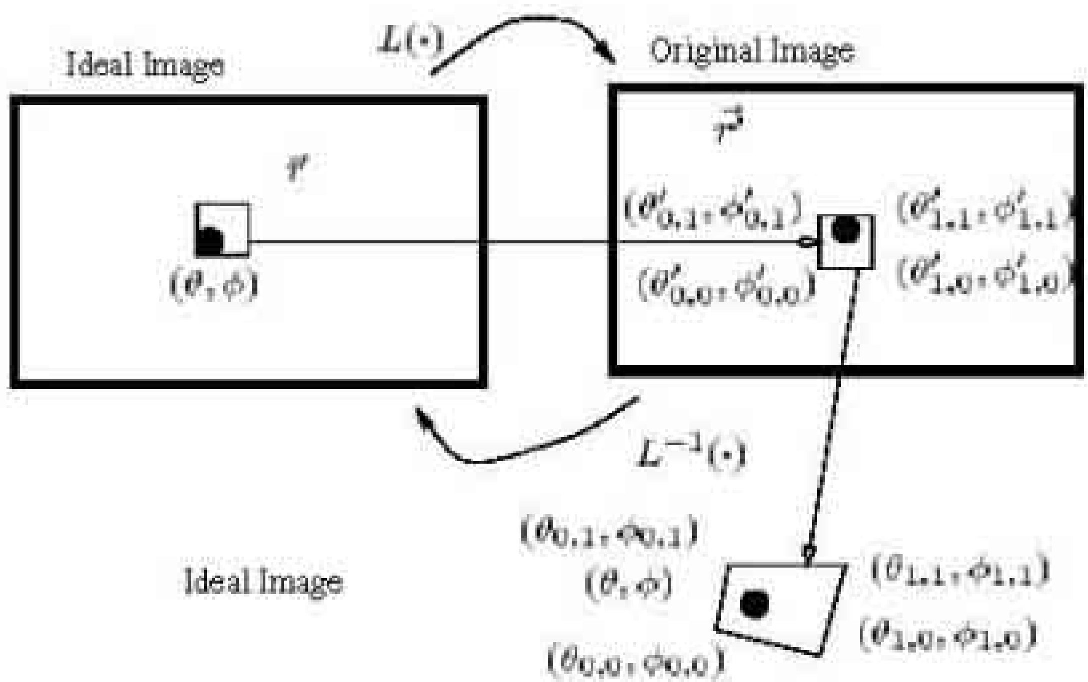
http://www.csl.sony.co.jp/person/nielsen/spatialmedia/

Acknowledgements: Sony Corporation.

EXTRA SLIDES

RAY INTERPOLATION (CONT'D)

Non ideal pinhole case (lens distortions)



Natural neighbor interpolation on the sphere

Watson, D.F., 1994, nngridr: An implementation of natural neighbor interpolation

EXAMPLES OF SYSTEM CONFIGURATIONS

- *Single camera head* (catadioptric systems)

Accowl Corp.

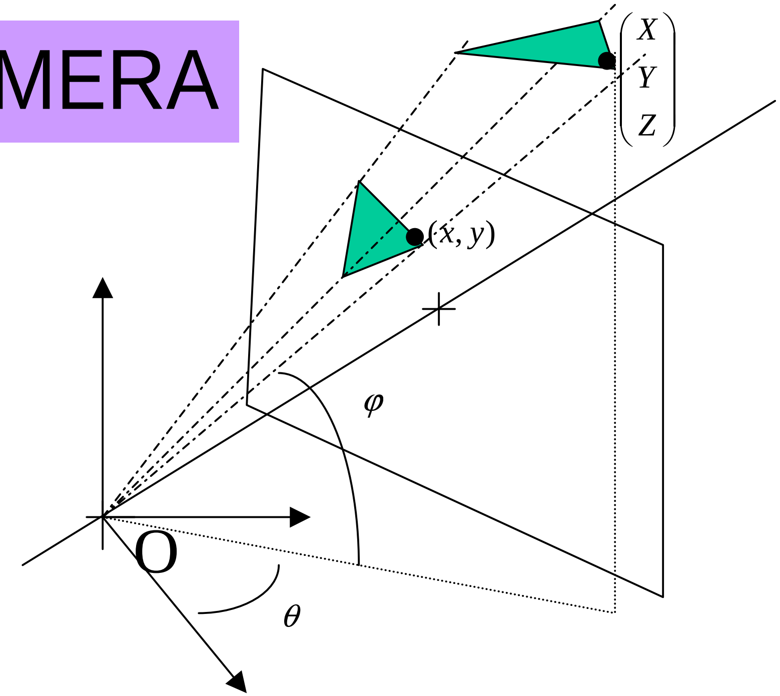
- *Multiple camera heads* (close nodal points)

- *Virtual nodal point alignments* (using mirrors)

- Etc.

EXAMPLE 1: PINHOLE CAMERA

$$\frac{x}{X} = \frac{y}{Y} = \frac{f}{Z}$$



$$\text{RayToImage}(\theta, \varphi, C) = (f \tan\theta, \sqrt{x^2 + f^2} \tan\varphi)$$

$$\text{ImageToRay}(x, y, C) = (\arctan\frac{x}{f}, \arctan\frac{y}{\sqrt{x^2 + f^2}})$$

EXAMPLE 2: FISHEYE CAMERA

$$\text{RayToImage}(\theta,\varphi,C) = (c_x + F^{-1}(\varphi)\cos\theta, c_y + F^{-1}(\varphi)\sin\theta)$$

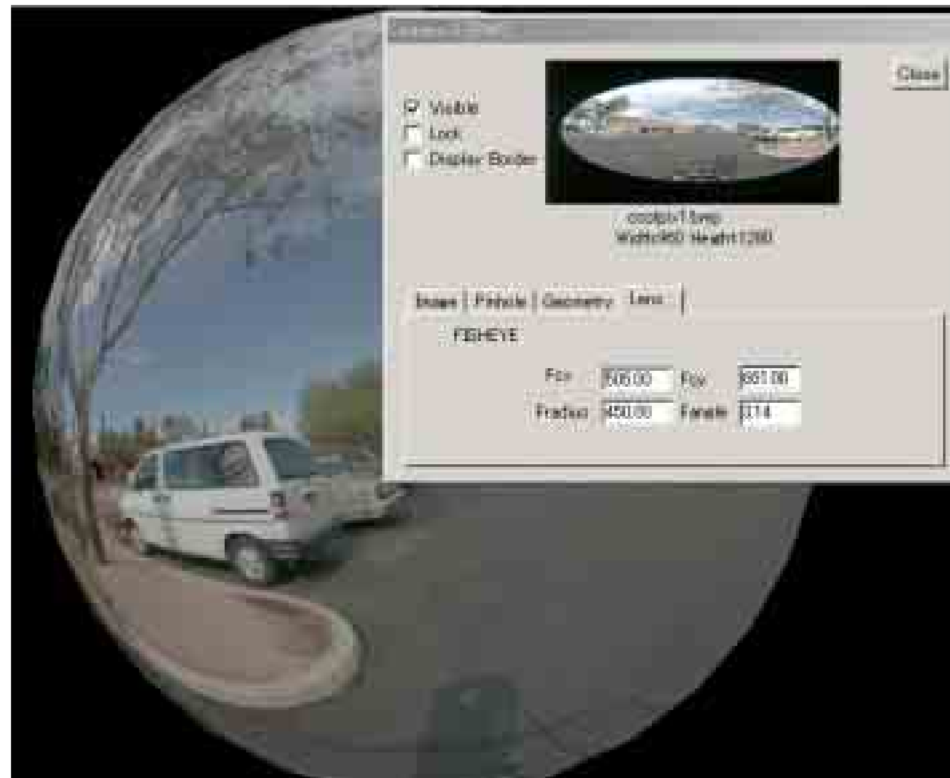
$$\text{ImageToRay}(x,y,C) = (\arctan\frac{y-c_y}{x-c_x}, F(\sqrt{(x-c_x)^2+(y-c_y)^2}))$$

with

$$F(r) = \frac{fov}{2} \times \frac{r}{r_{\max}}$$

and

$$F^{-1}(\varphi) = \frac{2r_{\max}}{fov}\varphi$$



(Equidistant projection model)

SONY CSL

EXAMPLE 3: ANOTHER PINHOLE CAMERA

$$K = \begin{bmatrix} f_x & 0 & \frac{w}{2} \\ 0 & f_y & \frac{h}{2} \\ 0 & 0 & 1 \end{bmatrix}$$

$$f_x = \frac{w}{2\tan\frac{hfov}{2}}, \text{horizontal field of view } hfov$$

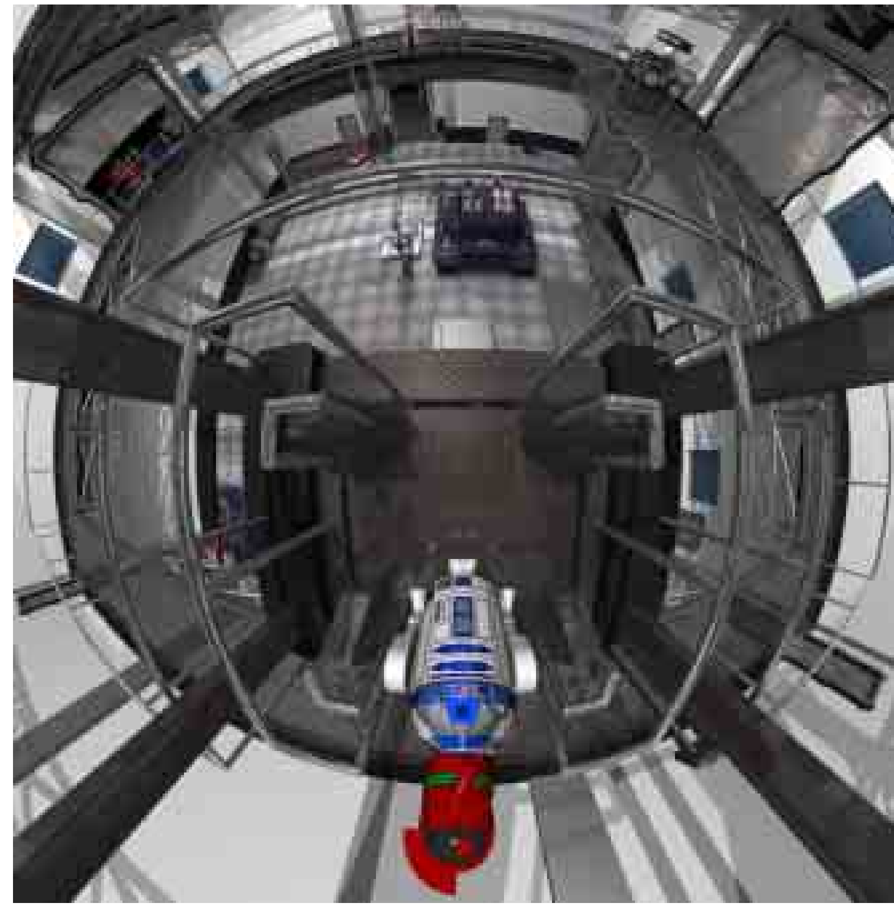
$$f_y = \frac{h}{2\tan\frac{vfov}{2}}, \text{vertical field of view } vfov$$

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = K^{-1} \times \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \rightarrow \text{Cartesian to spherical coordinates}$$

SONY CSL

SPATIAL MEDIA FOR COMPUTER GRAPHICS

➨dual paraboloid environment map of a CG script



(no stitching and all 3d and color information known)