THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

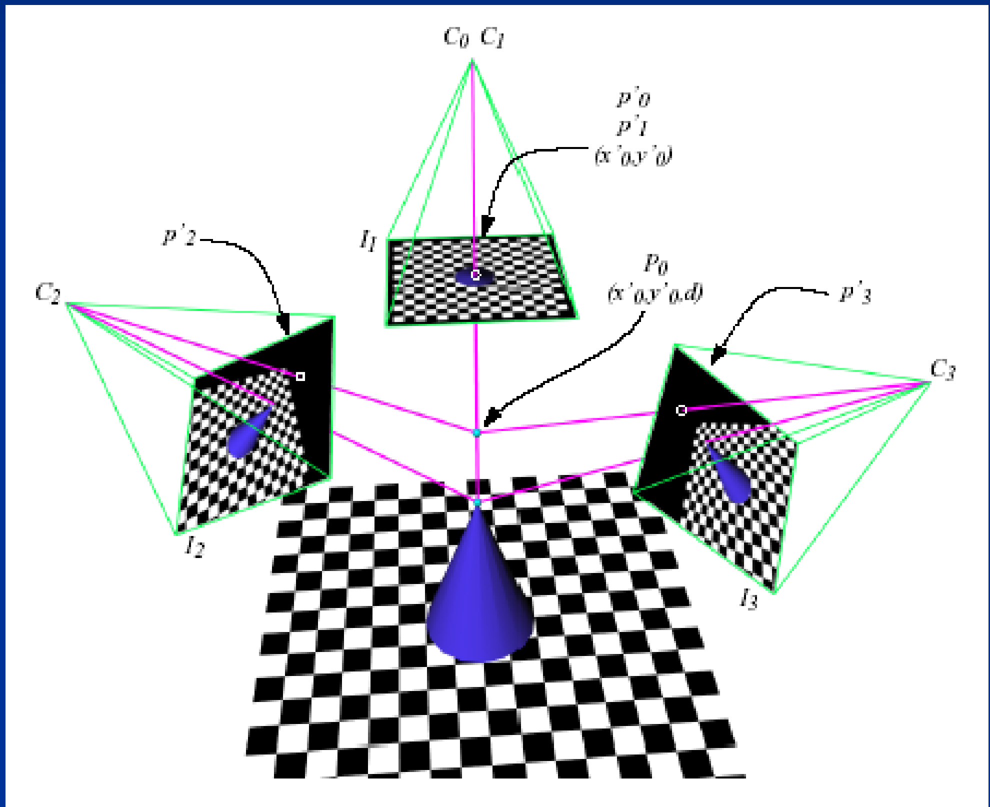
3D Reconstruction
from
Multiple View Images

Image Processing and Computer Vision

3D Reconstruction from Multiple View Images

- Review of 3D Reconstruction techniques
- Projective Geometry
- Volumetric Scene Modelling
  - Shape from Silhouette
  - Voxel Colouring
- Embedded Voxel Colouring
- Stereo Matching
  - Improving Speed
  - Improving Quality
- 4D Reconstruction from Image Sequences

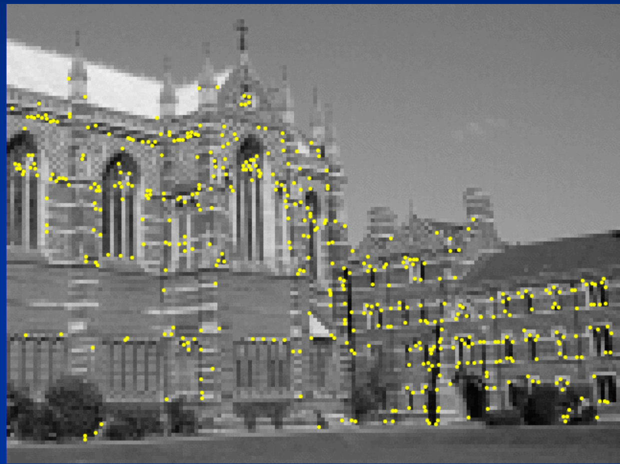
3D Reconstruction from Images



- 3D Reconstruction
  - Sparse
  - Dense
    - Image Correspondence
    - Volumetric Modelling

Aim: Recover the lost third dimension – Depth – from images alone

Sparse Reconstruction



3D Reconstruction

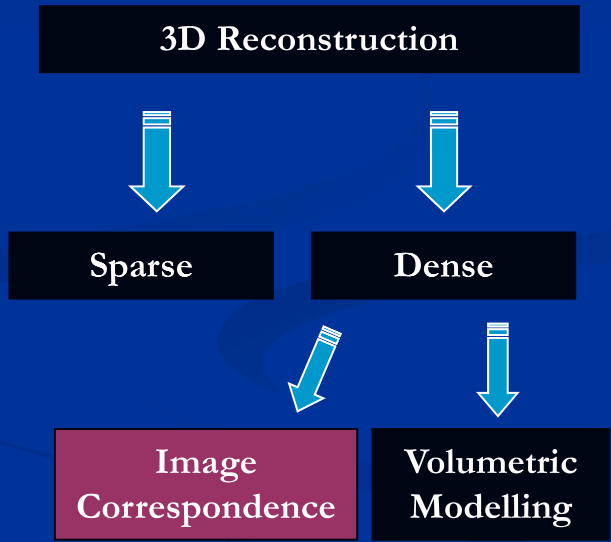
Sparse

Dense

Image Correspondence

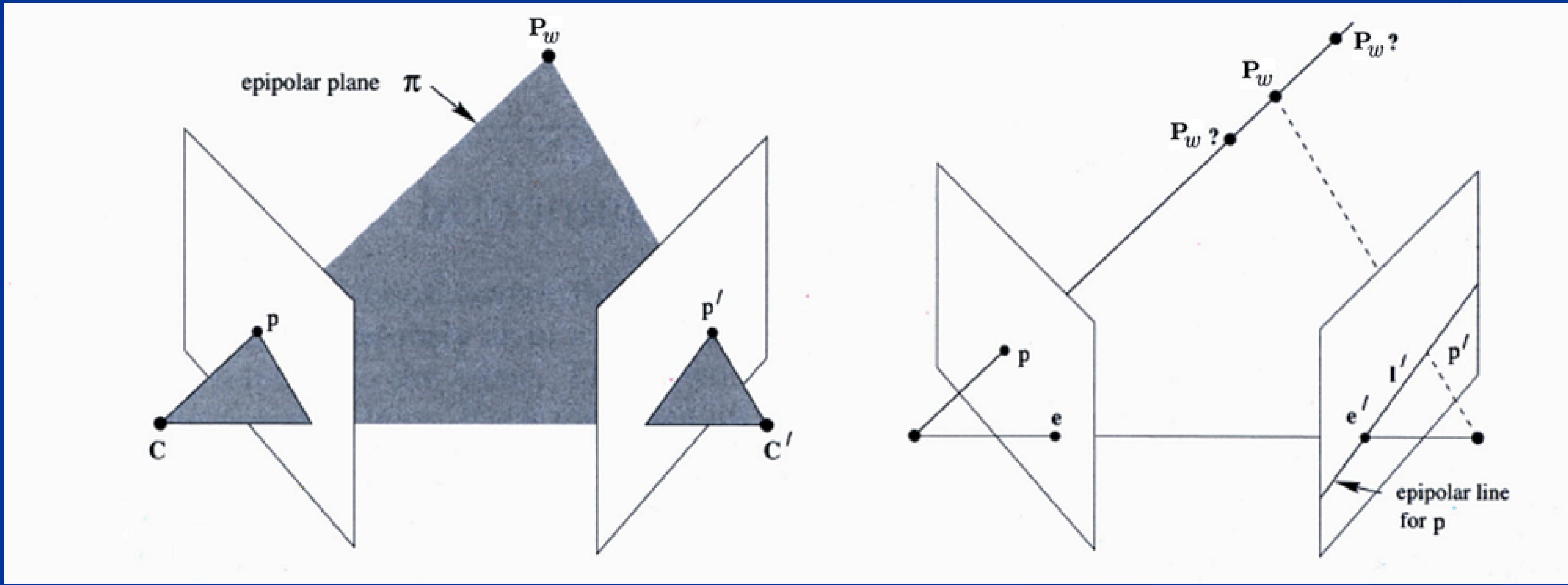
Volumetric Modelling

Dense Reconstruction : Feature Correspondence Problem

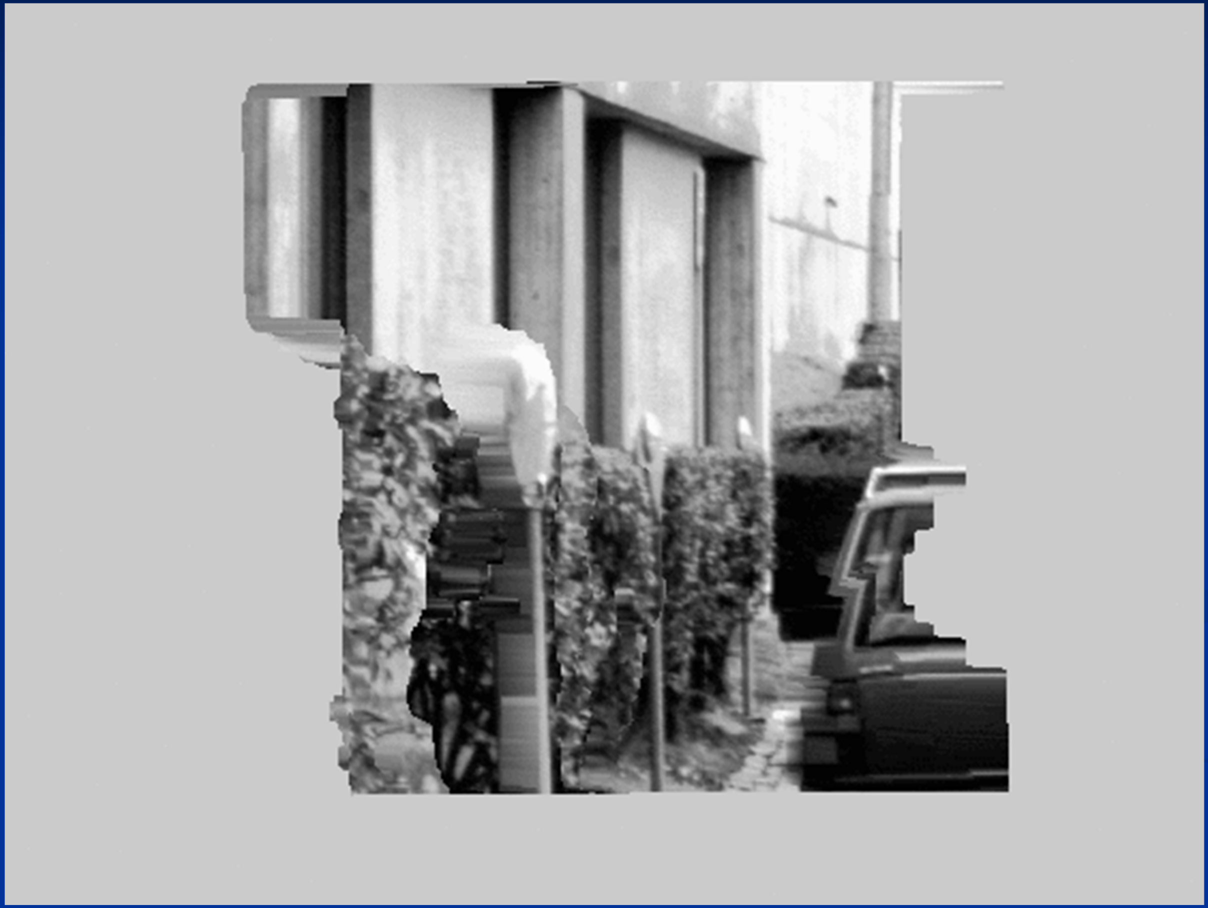


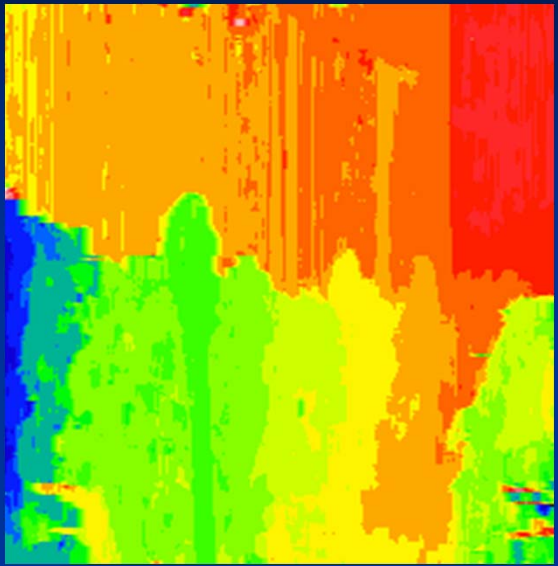
Stereo Matching

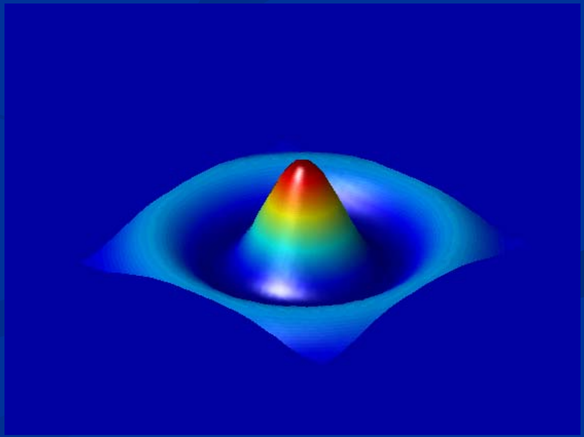
Epipolar Geometry



2.5D Sketch



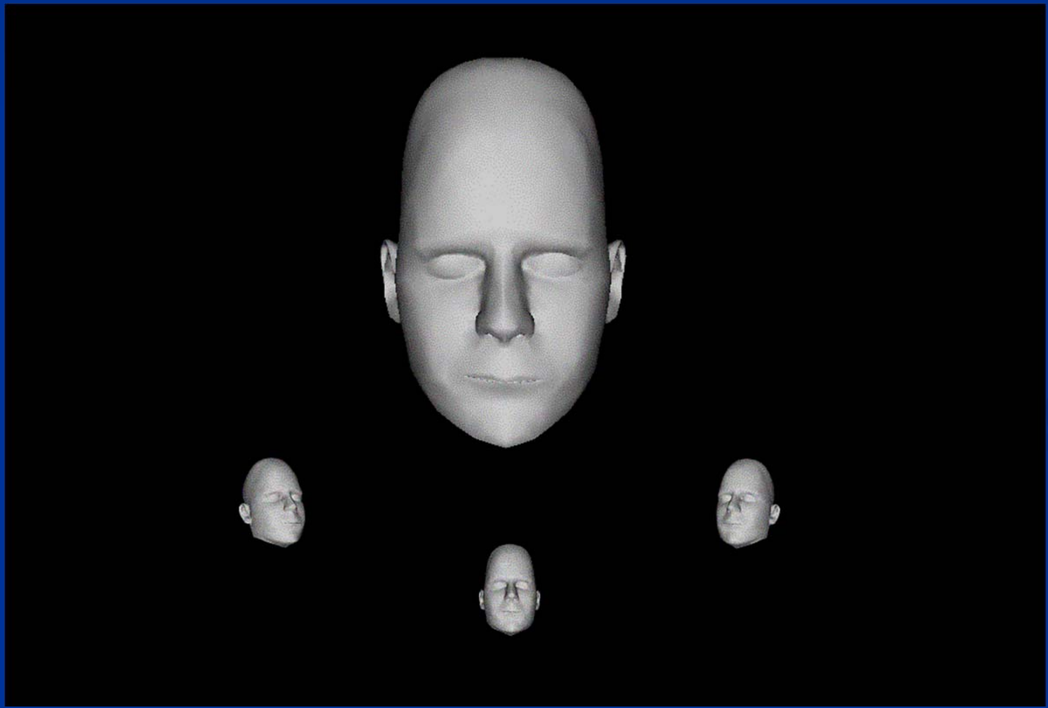




$$z = f(x, y)$$

Stereo Matching

3D Reconstruction from Multiple Views



3D Reconstruction

Sparse

Dense

Image Correspondence

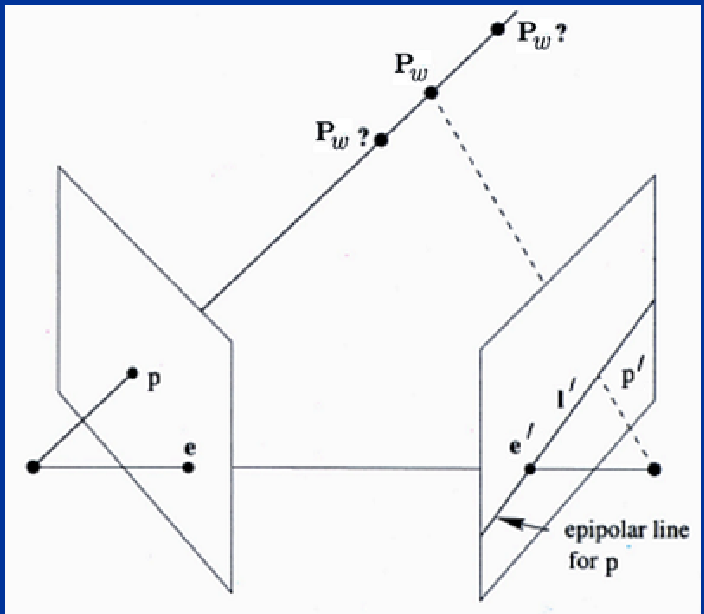
Volumetric Modelling

Projective Geometry

- Projective Coordinates

$$(x, y, w) \rightarrow \left(\frac{x}{w}, \frac{y}{w}\right)$$

$$(x, y, z, w) \rightarrow \left(\frac{x}{w}, \frac{y}{w}, \frac{z}{w}\right)$$



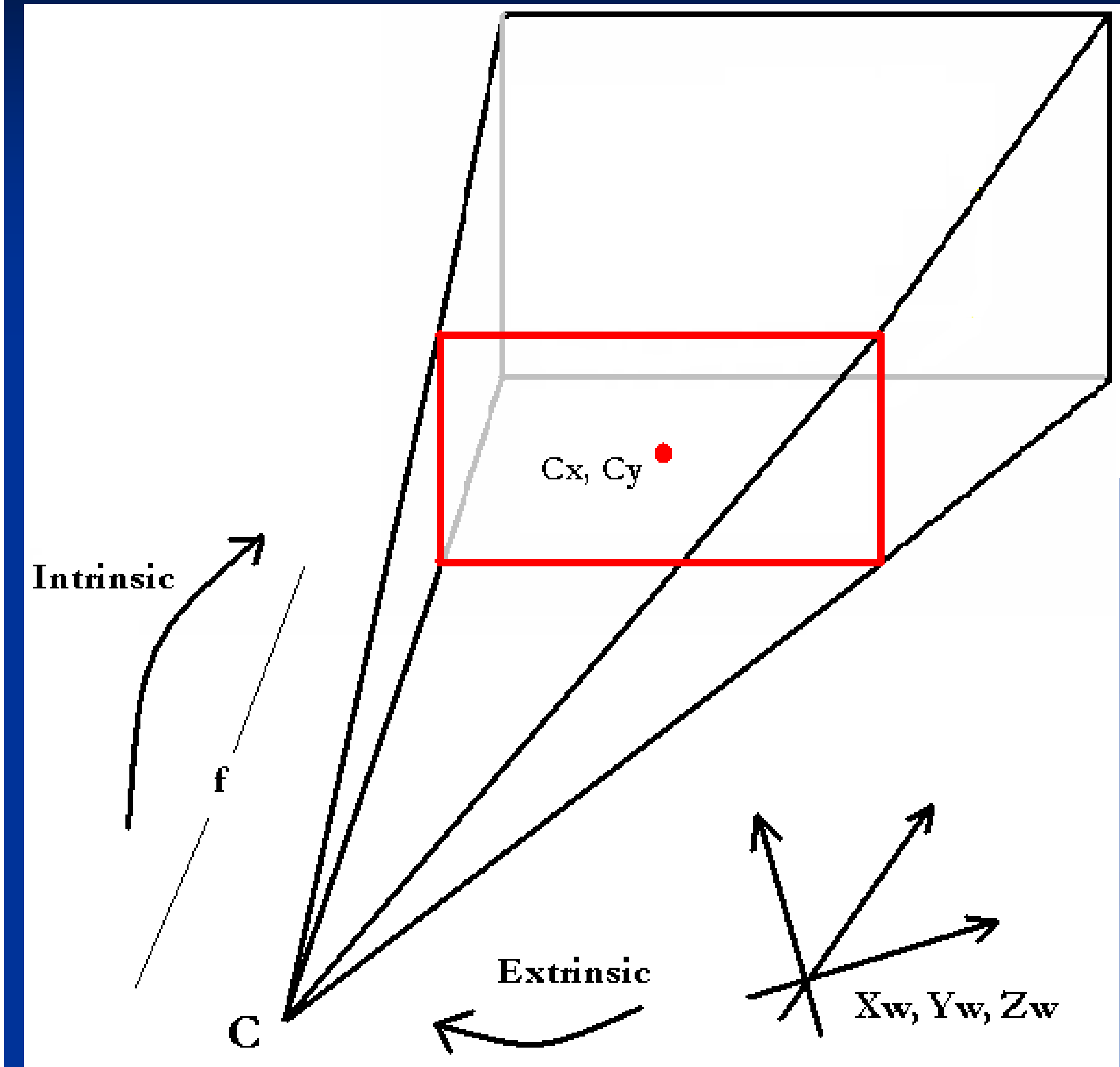
Epipolar Constraint:

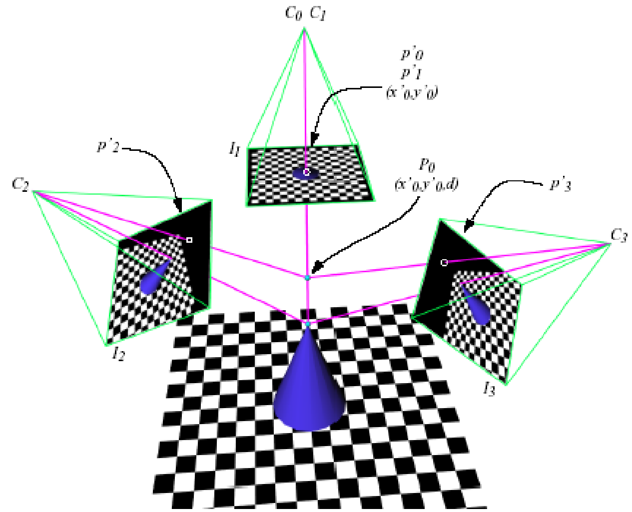
$$p'^T \mathbf{F} p = 0$$

F is a 3x3 Matrix

Calibration = estimate F

Projective Geometry



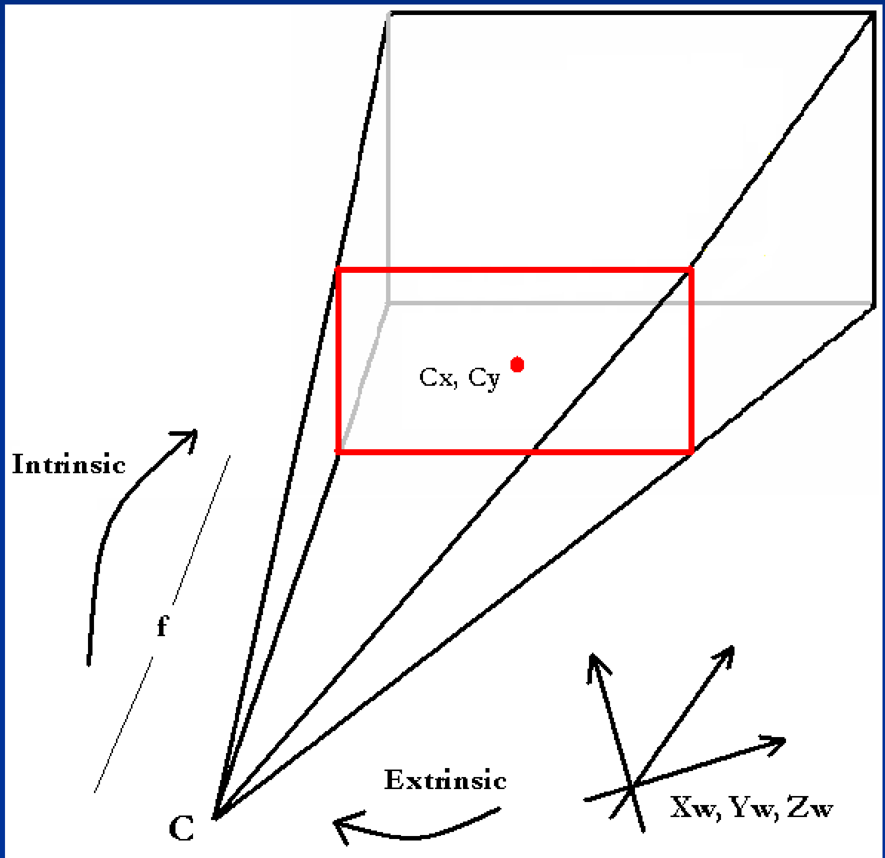


- Calibration is to find relationship:

$$(x, y, z, w) \leftrightarrow (x, y, w)$$

computing the Projection Matrix

Projective Geometry



Step 1: Compute

Extrinsic Transformation

$$\begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = R \begin{pmatrix} x_w \\ y_w \\ z_w \\ w_w \end{pmatrix} + T$$

$$\begin{bmatrix} R & T \\ 0^T & 1 \end{bmatrix}$$

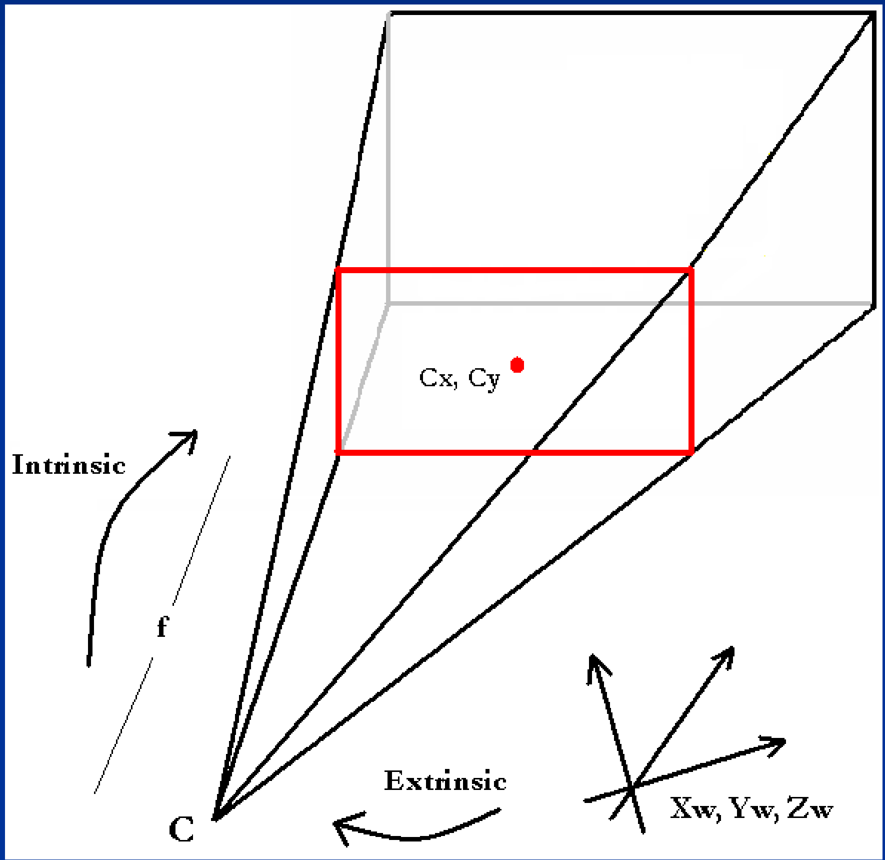
$$\begin{bmatrix} R & T \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ 0 \end{bmatrix}$$

Euclidean

$$\begin{bmatrix} R & T \\ v^T & v \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

Projective

Projective Geometry

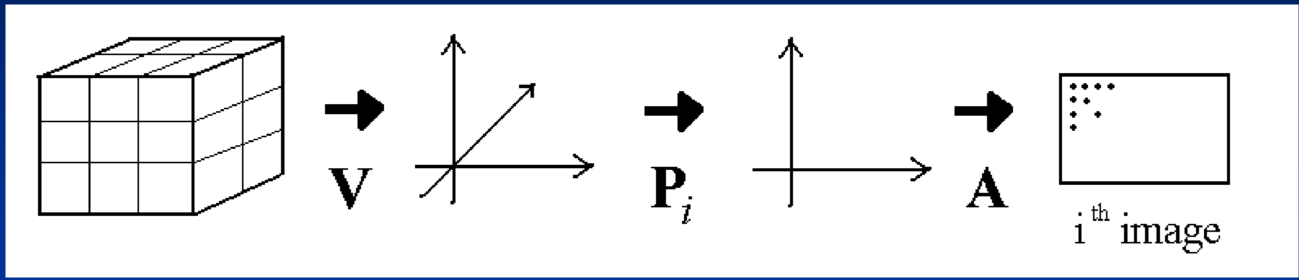


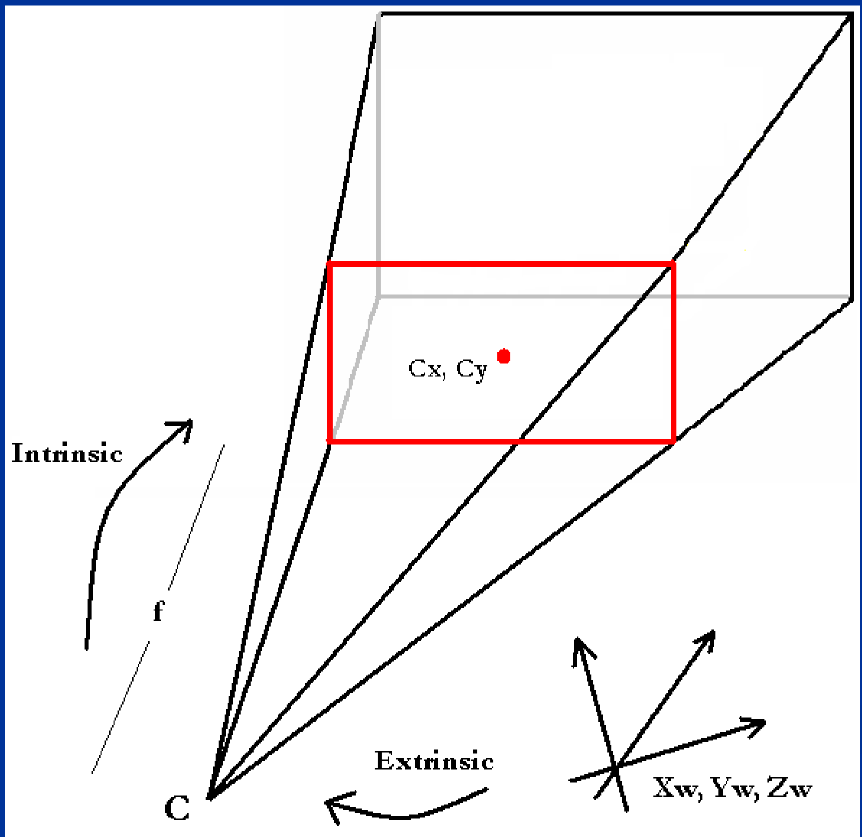
Step 2: Compute Projective Matrix

$$\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \\ w_i \end{bmatrix}$$

$$(x, y, z, w) \leftrightarrow (x, y, w)$$

Projective Geometry

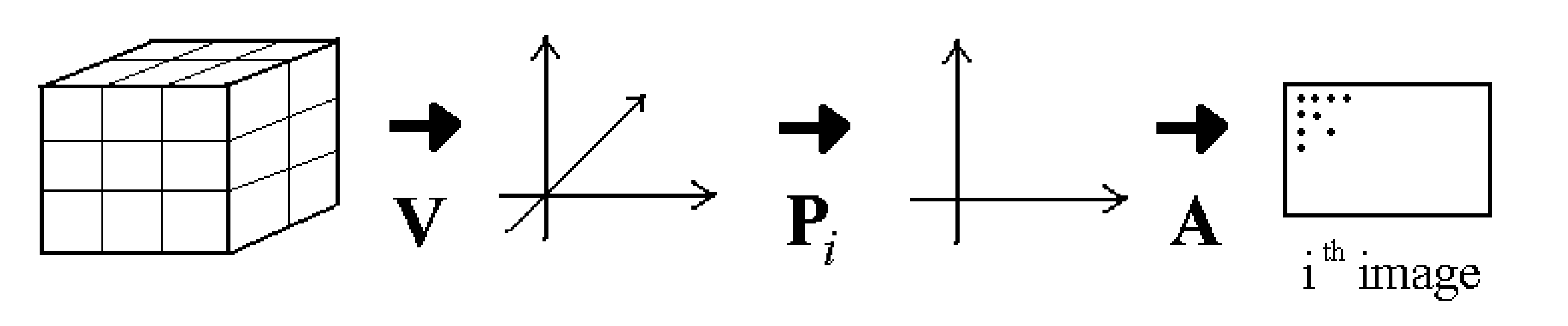




Step 3: Add in Intrinsic Transformation

$$A = \begin{bmatrix} f_x & s & C_x \\ 0 & f_y & C_y \\ 0 & 0 & 1 \end{bmatrix}$$

Projective Geometry



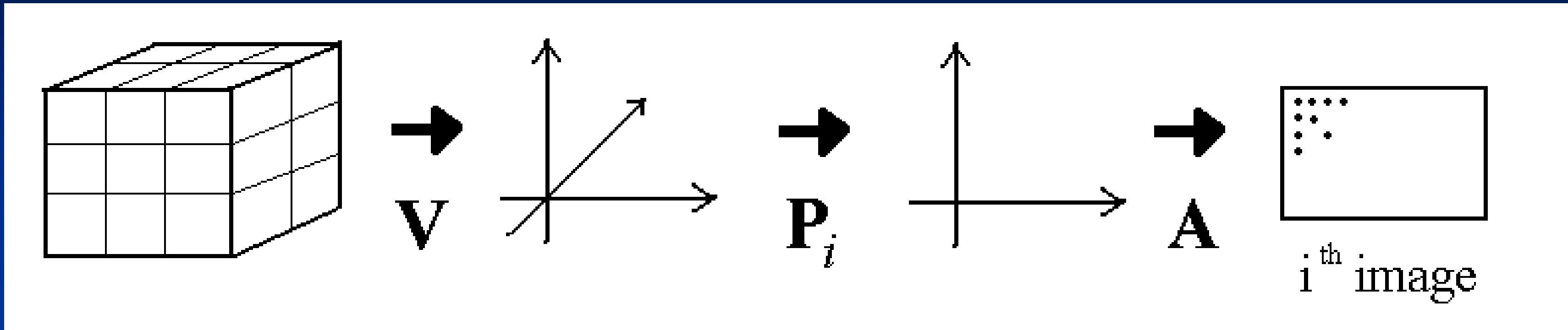
$$p_i = \mathbf{A}\ \mathbf{P}_\mathrm{i}\ \mathbf{V}\ v_m$$

A P$_\mathrm{i}$ = Projection Matrix, P

P = A [R | -RT]

$$P\begin{bmatrix} x_w \\ y_w \\ z_w \\ w_w \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \\ w_i \end{bmatrix}$$

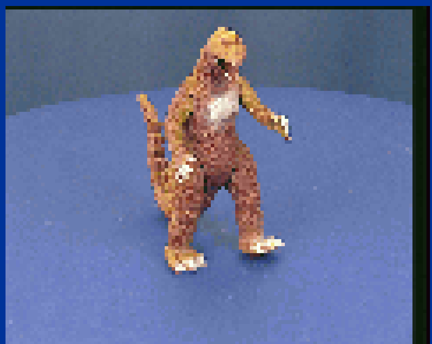
Projective Geometry

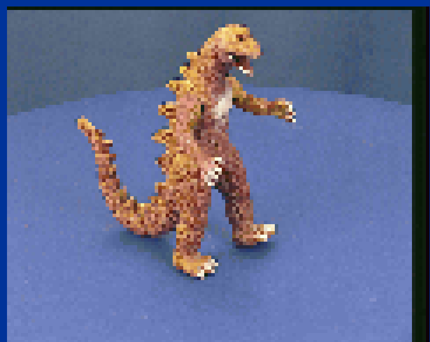


$$\boxed{p_i = \mathbf{A}\,\mathbf{P}_\mathrm{i}\,\mathbf{V}\,v_m}$$

- Estimating the 12 parameters of the Projection Matrix is a non-trivial task
- In your assignment, you are given the Projection Matrices = $\mathbf{A}\,\mathbf{P}_\mathrm{i}$
- Design V matrix to compute 3D coordinate of each voxel
- Region of Interest in world coordinate

Volumetric Modelling



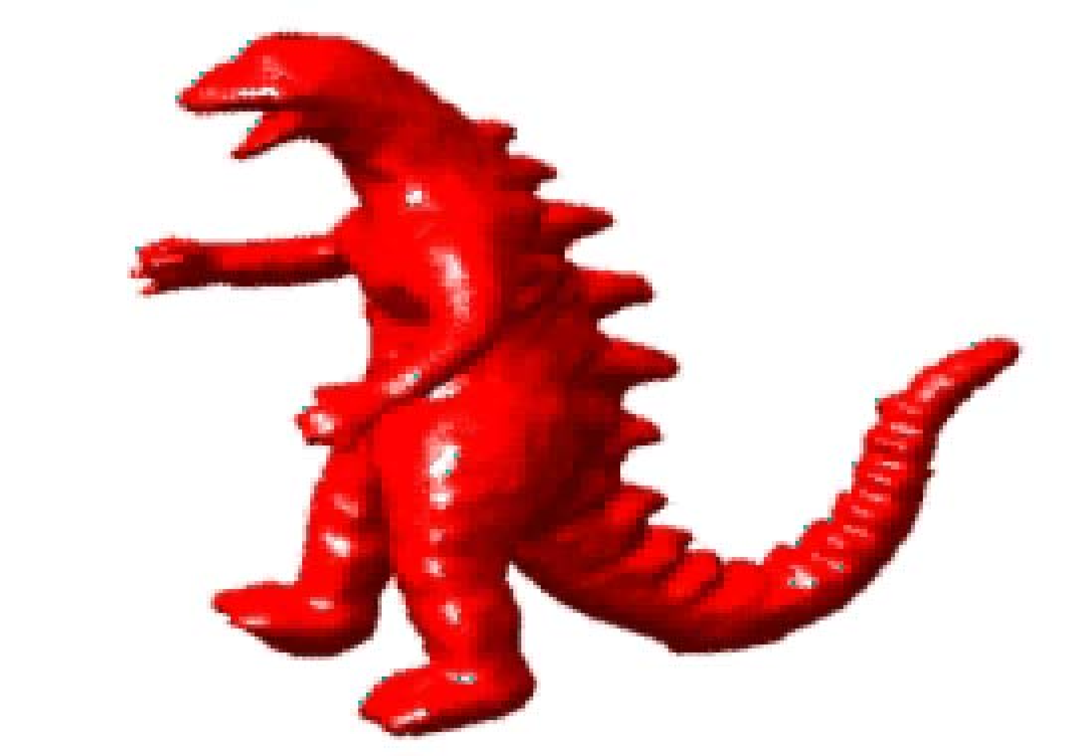


Shape from Silhouette

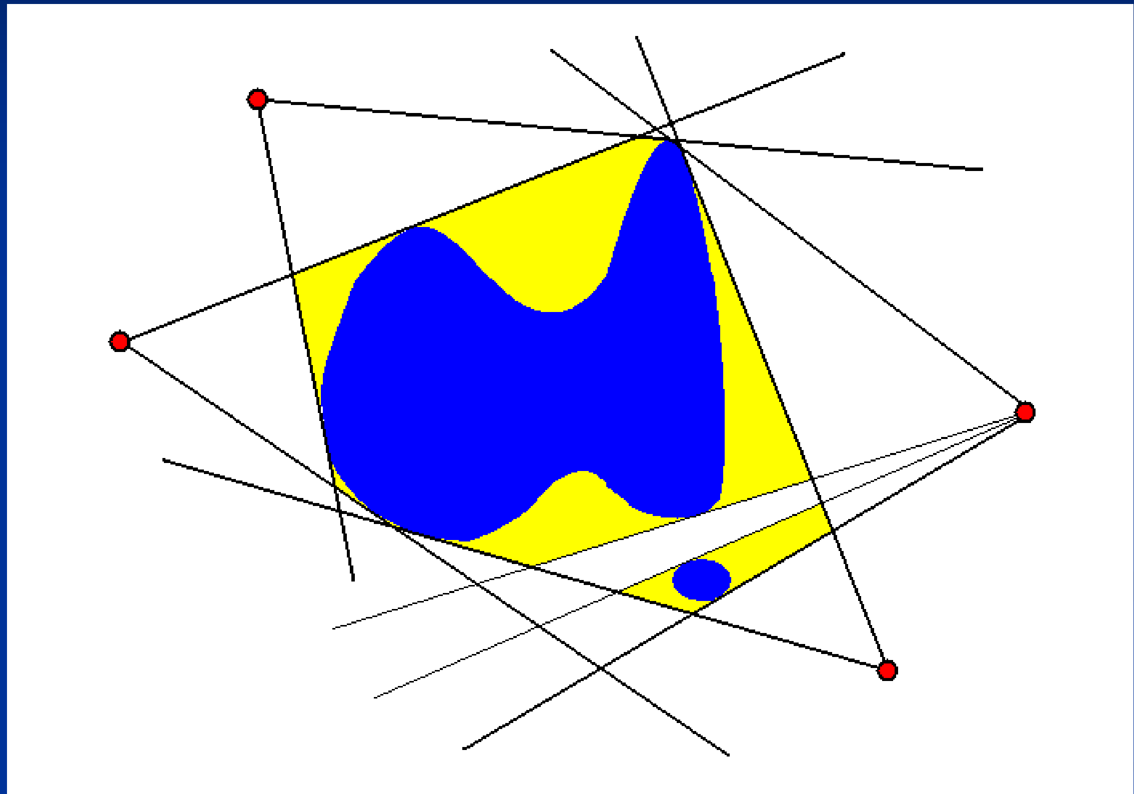
Shape from Silhouette

- Project the frustum of each silhouette and compute intersections
- Back-Project each voxel into all images and CARVE away non-dinosaur voxels

Shape from Silhouette

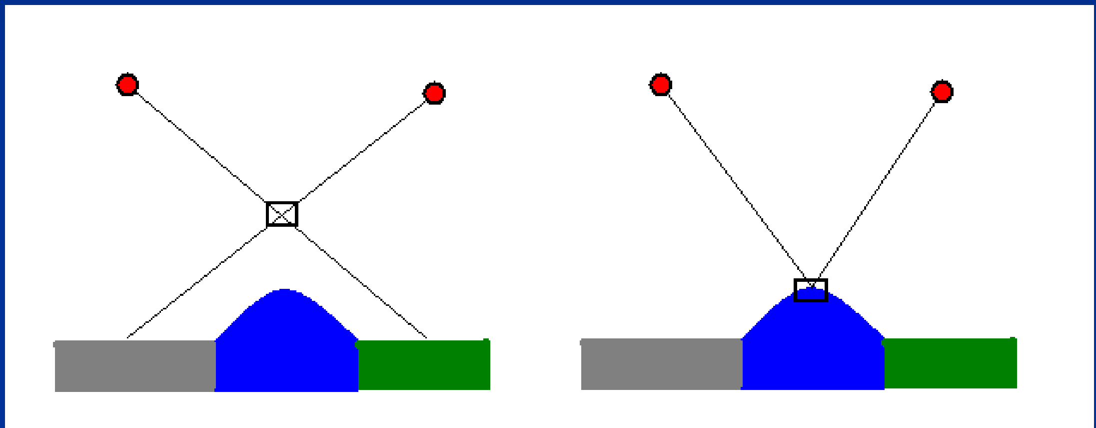


Shape from Silhouette



- Sensitive to Segmentation Errors (eg. Table extraction)
- Reconstruction by geometric intersection → Visual Hull

Shape from Photo-Consistency



Inconsistent voxels are *carved*

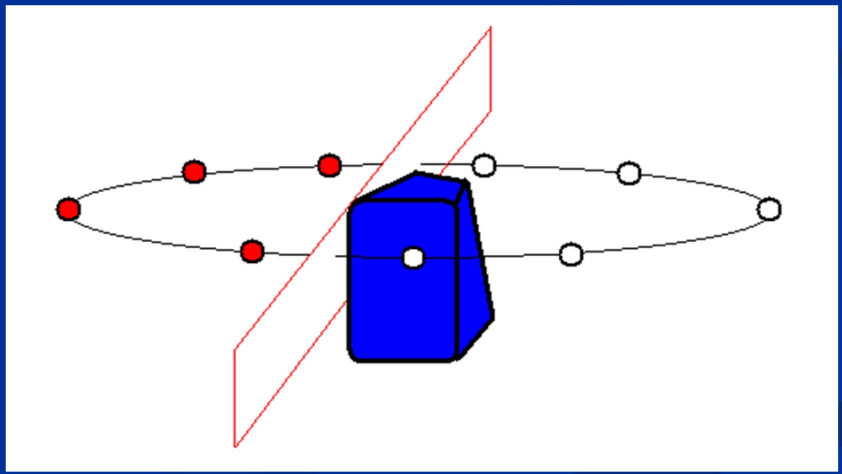
- Metric :
  - difference measure
  - variance
  - probability density function
  - histogram

Space Carving or Voxel Colouring

- S. Seitz and C. Dyer, “Photorealistic Scene Reconstruction by Voxel Coloring”, IJCV, Vol. 35, No. 2, 1999, pp. 151-173.

Occlusion Modelling

- Voxel Colouring
  - Ordinal Visibility Constraint – near to far traversal ordering
  - Camera location restricted
- Space Carving
  - Iterated voxel colouring
- Generalized Voxel Coloring
  - Arbitrary camera placement
  - Single sweep

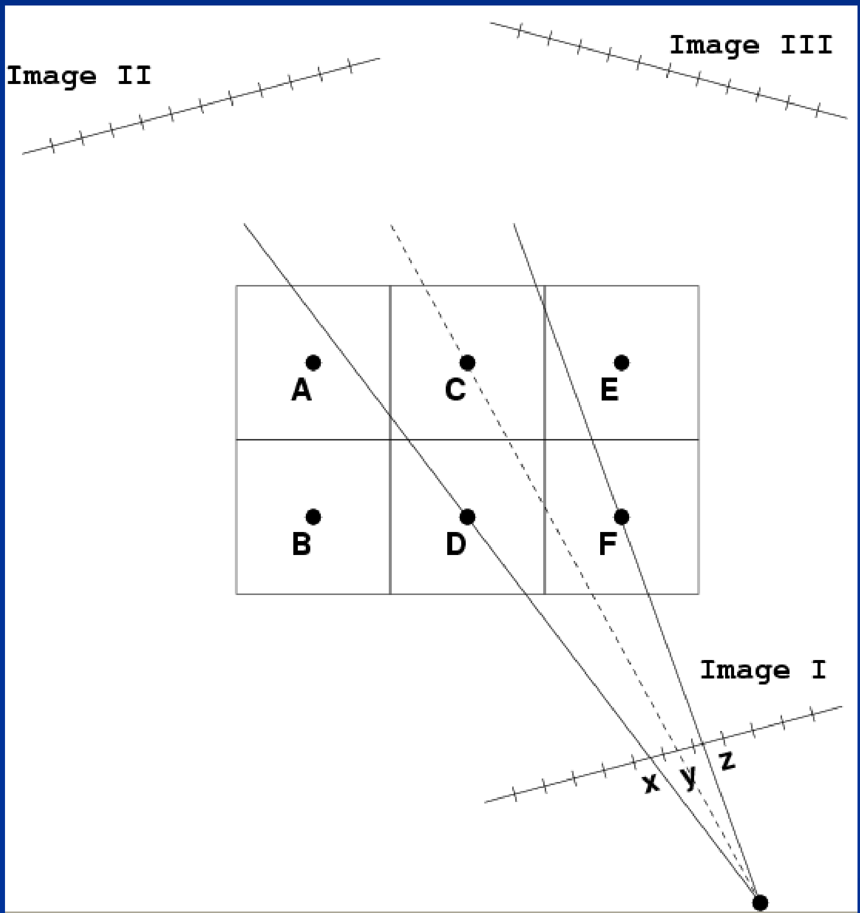


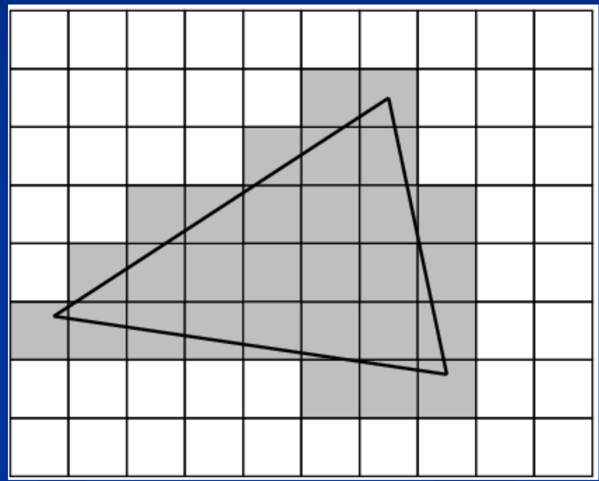
Embedded Voxel Colouring

- C. Leung, B. Appleton, C. Sun, *"Embedded Voxel Colouring"*, Digital Image Computing: Techniques and Applications, Vol. 2, pp. 623-632, December 2003.

- Properties of Carving
  - Water-Tight Surface Model
  - Monotonicity Carving Order
  - Causality

Water-Tight Surface Model





- Many voxels to many pixels relationship
- Water-Tight Voxels
- Water-Tight Pixels

Monotonic Carving Order

- Consider two carvings, S_A and S_B, computed at thresholds A and B. *Monotonicity of carving* dictates:

$$A \leq B \rightarrow S_A \subseteq S_B$$

- Therefore these sets may be embedded into a function!

$$S_A = \left\{ \mathbf{x} \middle| f\left(\mathbf{x}\right) \leq A \right\}$$

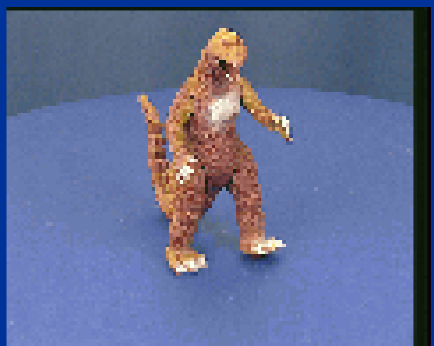
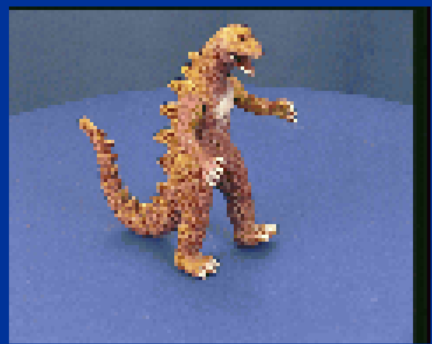
- Compute *f* in a single sweep
- All carvings may be obtained by thresholding

Causality

- Monotonic Carving Order + Water-tightness → Causality
- Under a water-tight surface model, only surface voxels get carved
- Every new surface voxel must have a neighbour who has been carved
- Every voxel has a neighbour of equal or higher consistency threshold
- No local maxima in the function f

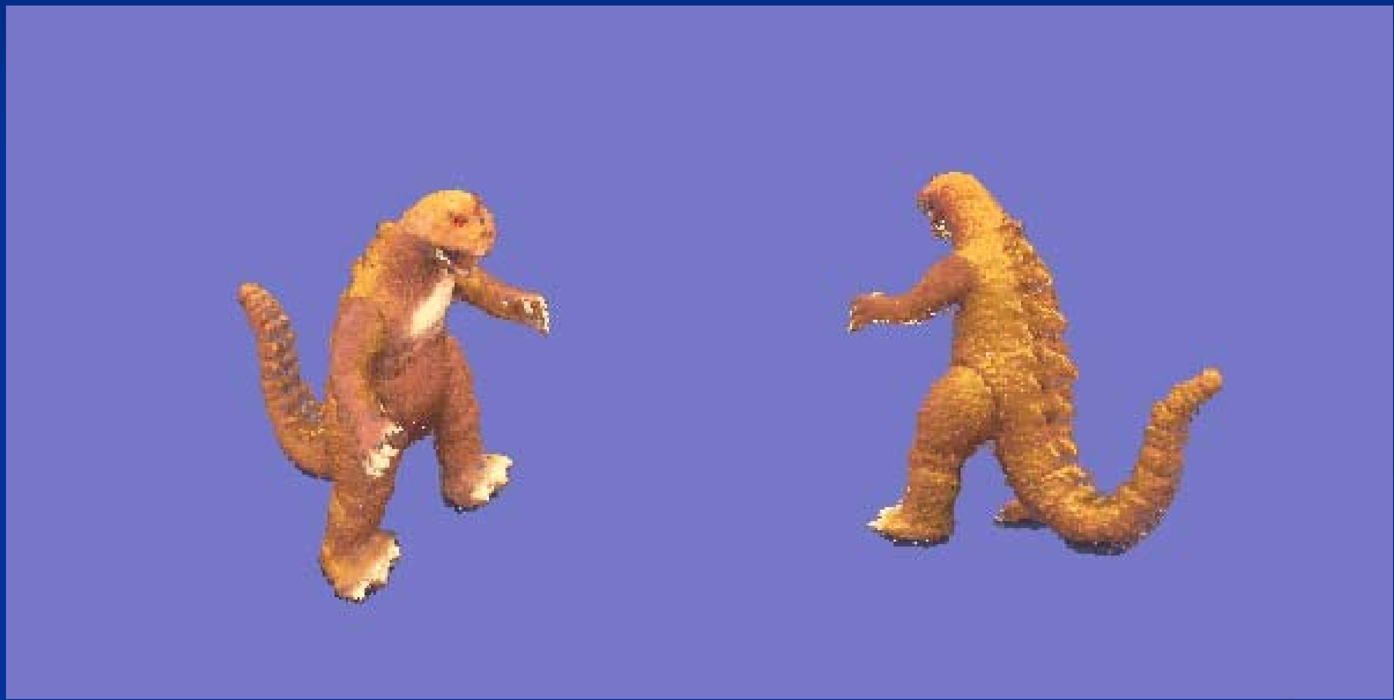
Volumetric Modelling



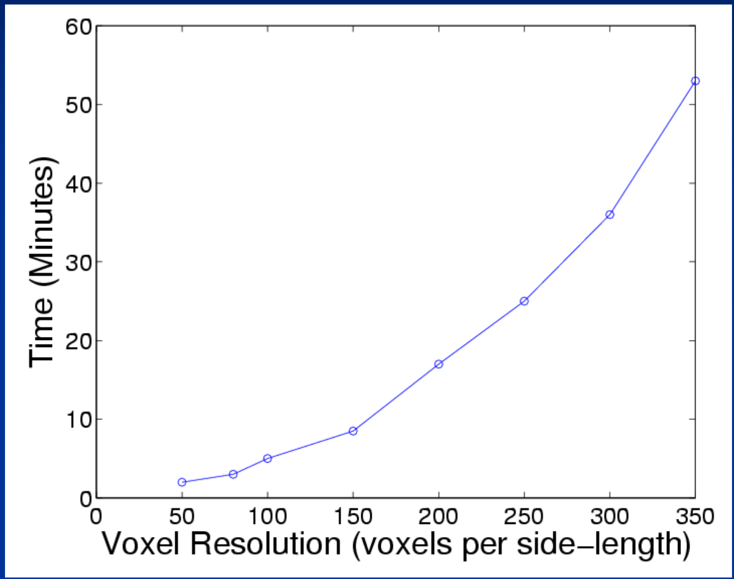

Results

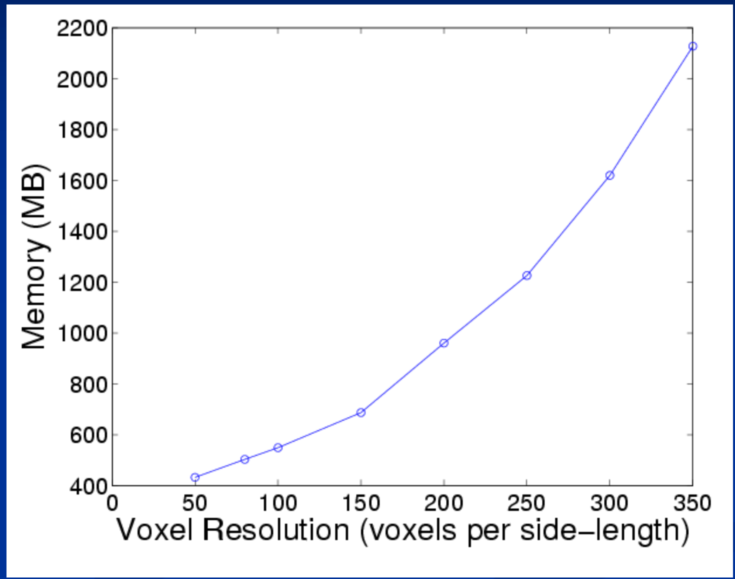
Embedded Voxel Colouring



- Embed carvings for all possible consistency threshold into one volume

Results





- Embedded VC :
  - 36 images (720x576)
  - 350x350x350 volume
  - 53 minutes (450MHz Ultra Sparc II)
- Generalised VC : (Culbertson et al.)
  - 17 images (800x600)
  - 167x121x101 volume
  - 40 minutes (440MHz HP J5000)

Stereo Matching

Multiscale

Box Filtering

$$\begin{bmatrix} 3 & 7 & 4 & 9 & 2 & 1 & 0 & 5 & 4 & 6 \end{bmatrix}$$

$$\begin{bmatrix} 3 & 7 & 4 & 9 & 2 & 1 & 0 & 5 & 4 & 6 \end{bmatrix}$$

Summing window of size 4 -

7 additions of a window size of 4

$$\begin{bmatrix} 23 & 22 & 16 & 12 & 8 & 10 & 15 \end{bmatrix}$$

Box Filtering

$$\begin{bmatrix} 3 & 7 & 4 & 9 & 2 & 1 & 0 & 5 & 4 & 6 \end{bmatrix}$$

Compute Accumulated Sum -

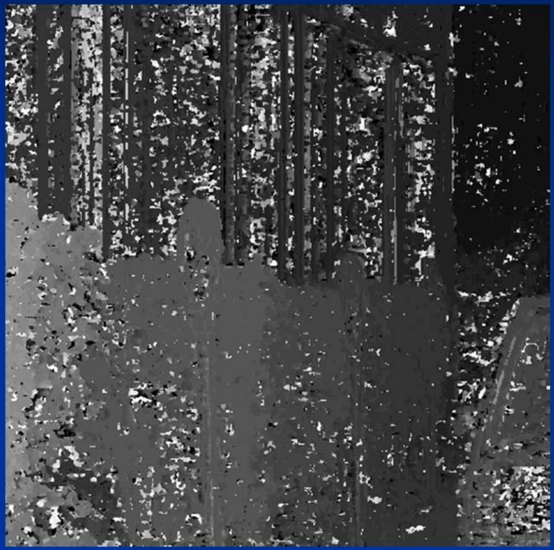
$$\begin{bmatrix} 3 & 10 & 14 & 23 & 25 & 26 & 26 & 31 & 35 & 41 \end{bmatrix}$$

Take Differences to obtain same result

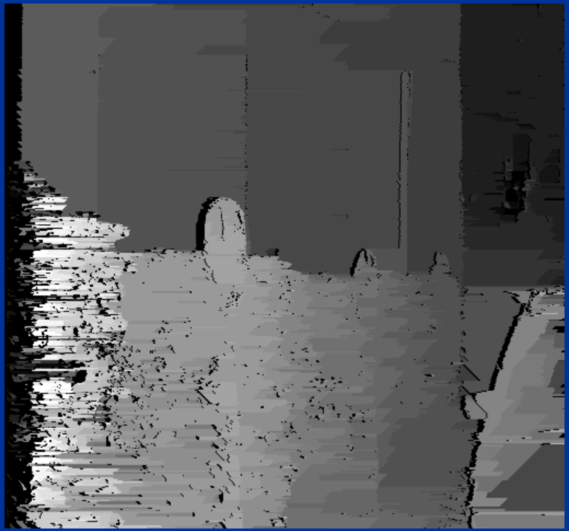
$$\begin{bmatrix} 23 & 22 & 16 & 12 & 8 & 10 & 15 \end{bmatrix}$$

Smoothness Constraint

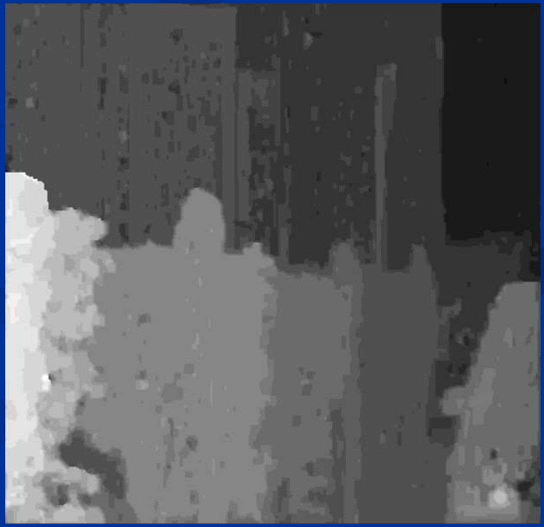
Greedy



Dynamic
Programming

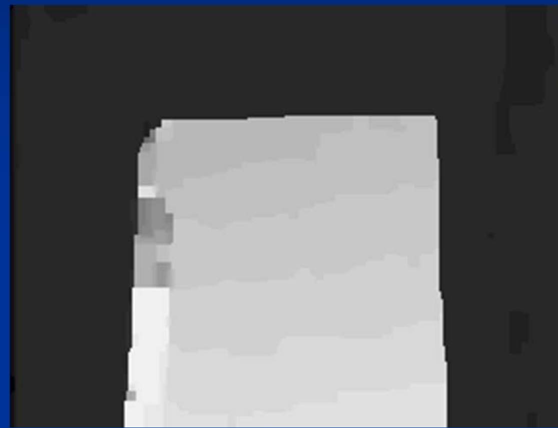


Iterated
Dynamic
Programming



Stereo Reconstruction using Iterated Dynamic Programming

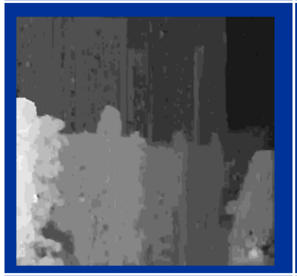
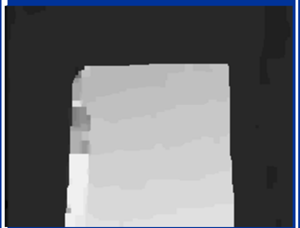
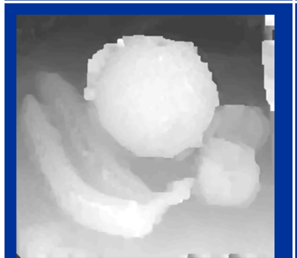
Ground Truth



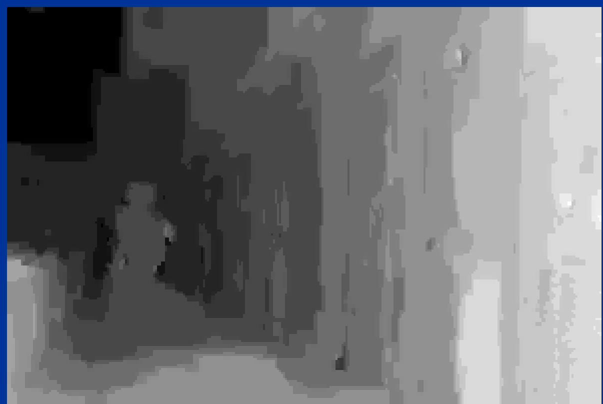
IDP

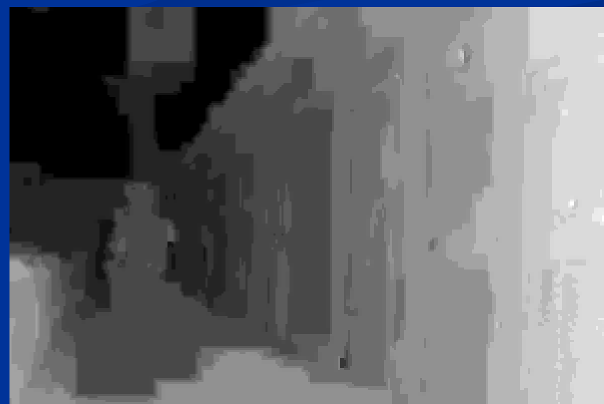
IDP

Stereo Reconstruction using Iterated Dynamic Programming and Quadtree Subregioning

Image	Size	Scales	Disparity range	Window size	Time (seconds)
	512×480	3	-30, 0	5×5	3.28
	284×216	1	-30, 0	3×3	1.1
	512×512	3	-25, 20	9×9	5.9

Stereo-Temporal Reconstruction
(3.5D Reconstruction)

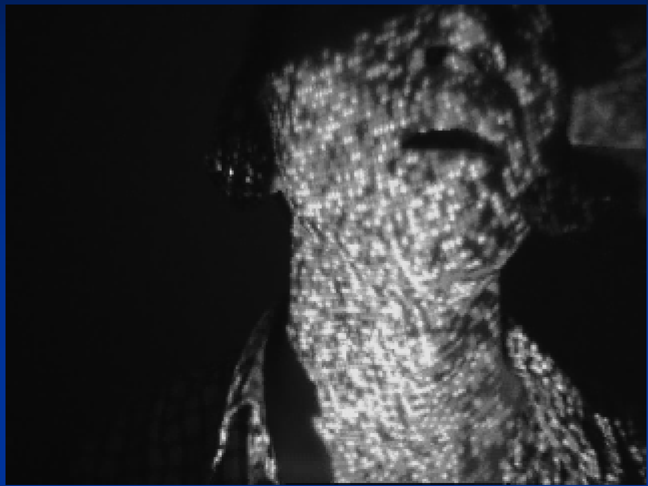




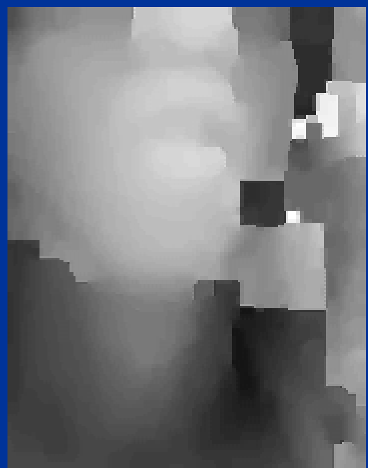
Without Temporal Coherence

With Temporal Coherence

Stereo-Temporal Reconstruction

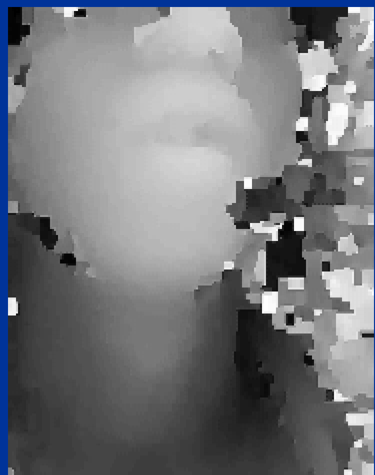


Without Temporal

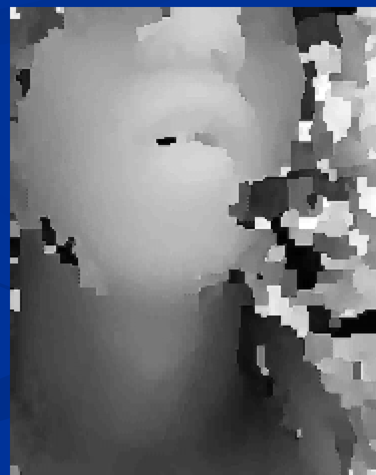


With Temporal

5×5 window, $K_2 \approx K_1$



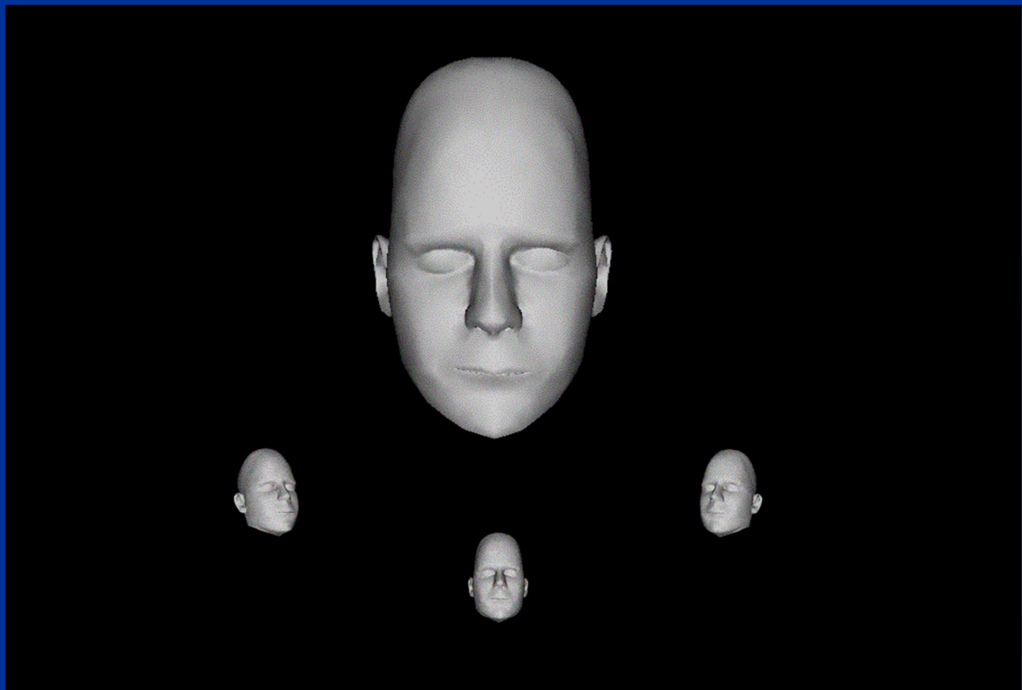
Without Temporal

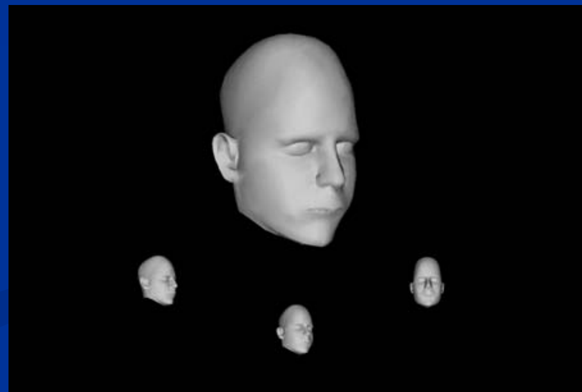


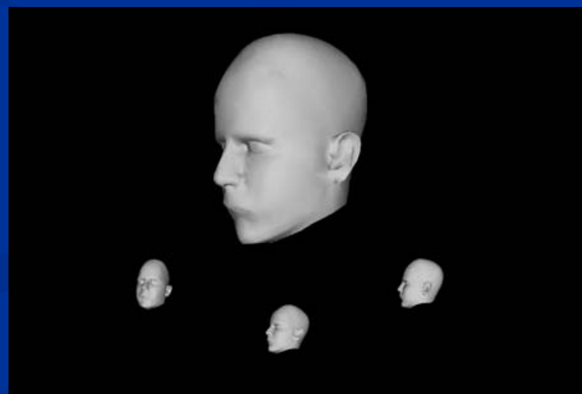
With Temporal

3×3 window, $K_2 > K_1$

3D Dynamic Scene Reconstruction from Multiple View Image Sequences
(4D Reconstruction)







3D Reconstruction from Multiple View Images