Recovery of 3-Dimensional Scene Information from Video Images

Tardi Tjahajdi
Image Processing and Expert Systems Laboratory
Information and Communication Research Group
School of Engineering
University of Warwick

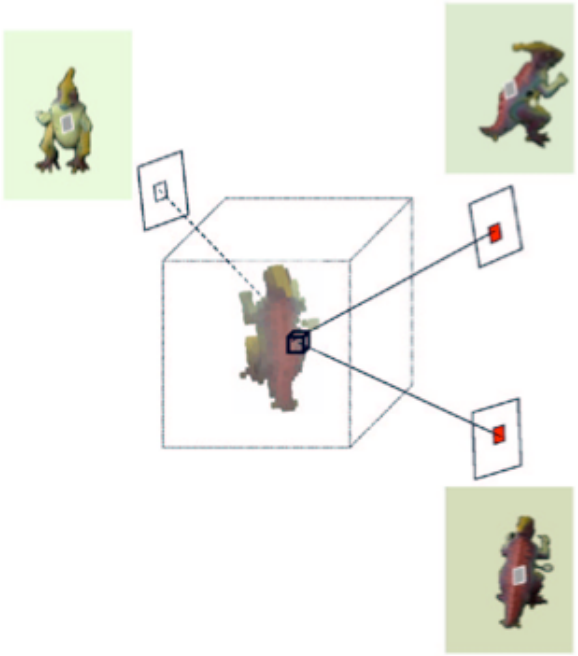
THE UNIVERSITY OF WARWICK

Contents of the Talk

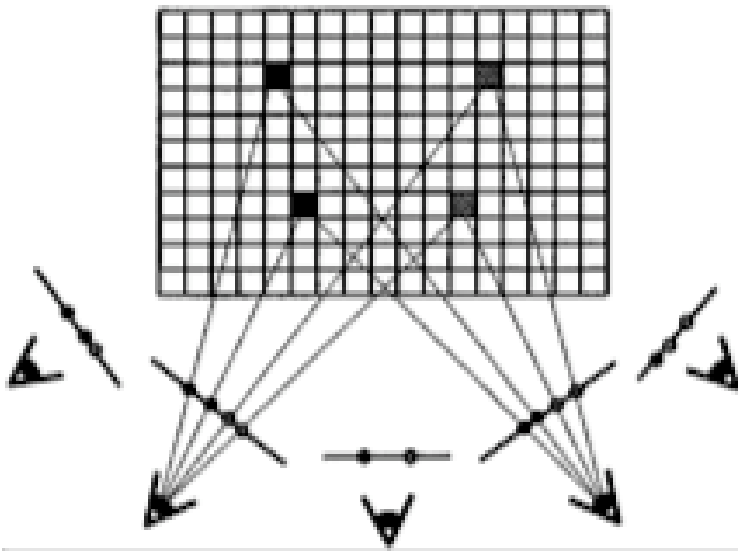
- Brief review of 3D object/scene reconstruction
- Camera calibration
- Structure from motion
- Shape from Silhouette

3D Object Reconstruction Methods

- Applications: robot navigation, obstacle recognition, digital preservation of works of arts, etc.
- Active vision: laser scanning, stereo with structured lighting
- Volumetric reconstruction:



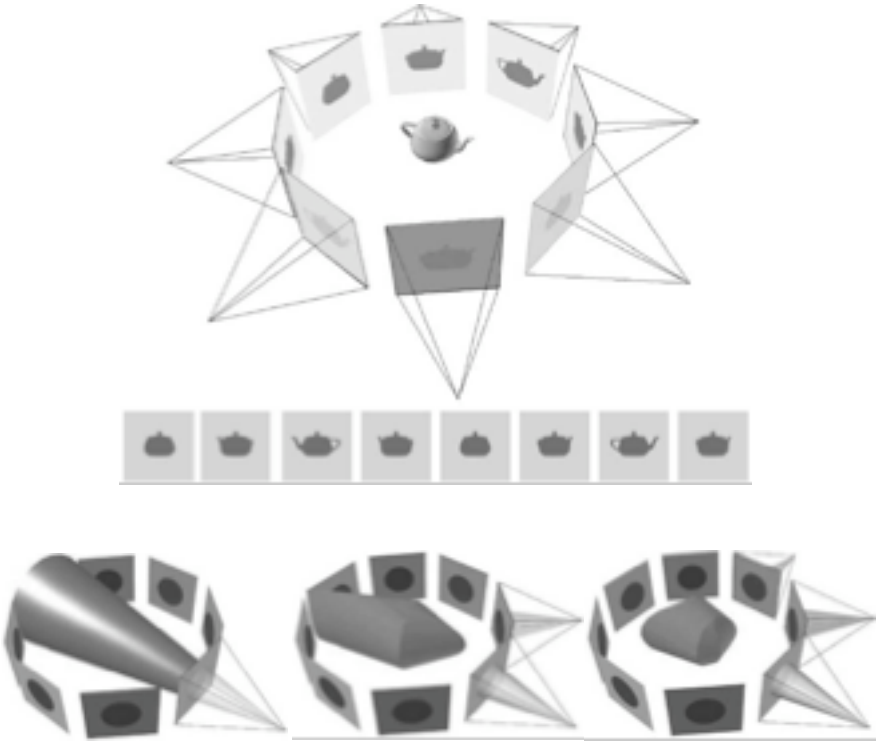
Space carving (Kutulakos & Seitz, 2000)



Voxel colouring (Seitz & Dyer 1999)

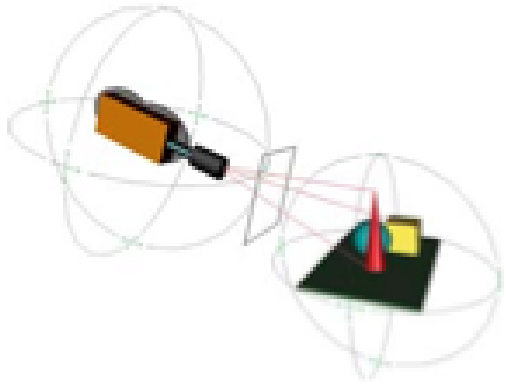
3D Object Reconstruction Methods (continued)

- Shape from stereo
- Shape from Silhouette (SfS)



SfS (Yemez & Schmitt, 2004)

- Structure from motion (SfM)



SfM (Jebara, Azarbayejani Pentland,1999)

Camera calibration

- Determines camera parameters
- Methods 1 using calibration object
  - e.g., Tsai, 1987; Zhang, 2000
- Methods 2 using geometric invariance of image features
  - e.g., Meng & Hu, 2003
- Methods 3: self-calibration
  - e.g., Pollefeys & van Gool, 1999

- Precision of calibration depends:
  - feature detection
  - camera model
- Parametric distortions
  - parametric deviations from pinhole model: methods 1
  - parametric models for specific lenses
    - fish-eye transform model, e.g., Basu & Licardie, 1995.
    - field-of-view model, e.g., Devermay & Faugeras, 2001.
- Generic (non-parametric) distortions
  - e.g., Mohr, Boufama & Brand, 1993

Multiview camera-calibration framework for nonparametric distortions removal (Vincent & Tjahjadi, IEEE Trans Robotics, 2005)

- The mapping between a 3D point $P_4 = [X,Y,Z,1]^T$ in an arbitrary world coordinate system and its ideal homogeneous image coordinates $p = [x,y,1]^T$ is

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \simeq \begin{bmatrix} f_x & \alpha & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T \quad (1)$$

  - where **R** is 3x3 projection matrix, **t** the translation vector, f_x and f_y the horizontal & vertical scale factors, α the image skew, and (u_o, v_o) the principal point.
- Taking into account of potential deviations,

$$\mathbf{p} = \begin{bmatrix} x & y & 1 \end{bmatrix}^T \simeq \mathbf{H} \begin{bmatrix} X & Y & 1 \end{bmatrix}^T = \mathbf{HP} \quad (2)$$

  - where H is 3x3 matrix corresponding to homography
- For a multiview calibration framework we need to minimize for all views the differencess between the transformations to obtain a common mapping for all views

- Define 2 affine transformations A and A' for p and P such that (2) becomes

$$\mathbf{Ap} \simeq \mathbf{AHA}'^{-1}\mathbf{A}'\mathbf{P} = \mathbf{H}'\mathbf{A}'\mathbf{P} \qquad (3)$$

  - where $\mathrm{H} = \mathrm{A}^{-1}\mathrm{H}'\mathrm{A}'$
- For distortion corrections, use a virtual grid defined by *P* calibration feature points P_i. For each view compute the best homography H^{j} which minimises the residual error associated with view j

$$\varepsilon^j = \min_{\mathbf{H}^j} \sum_i \|\tilde{\mathbf{p}}_i^j - \mathbf{H}^j\mathbf{P}_i\|^2 \qquad (4)$$

  - where $\tilde{\mathrm{p}}_i^j$ are the observed sub-pixel coordinates of P_i
- For each view, the corrected (undistorted) points $\tilde{\mathrm{p}}_i^{\mathrm{H}^{\mathrm{j}}}$ is given by

$$\hat{\mathbf{p}}_i^{\mathbf{H}^j} = \mathbf{H}^j\mathbf{P}_i \qquad (5)$$

- The estimations are further refined by fusing the corrective distortion maps obtained from different views into a common distortion map, applying this map to correct the distorted pixel coordinates.
- The corrective distortion vector is

$$\mathbf{c}_i^j = \begin{bmatrix} \sum_{k,l} B_{Hz}^{k,l} * \mathcal{N}^k(u(\tilde{p}_{ix}^j)) * \mathcal{M}^l(v(\tilde{p}_{iy}^j)) \\ \sum_{k,l} B_{Vz}^{k,l} * \mathcal{N}^k(u(\tilde{p}_{ix}^j)) * \mathcal{M}^l(v(\tilde{p}_{iy}^j)) \\ 1 \end{bmatrix} \quad (6)$$

  - where $B_{Hz}^{k,l}$ and $B_{Hz}^{k,l}$ are sets of control vectors, $N^k(u(\tilde{p}_{ix}^j))$ & $M^k(u(\tilde{p}_{iy}^j))$ are B-spline basis function in *u* & *v* directions respectively
- The best B-spline surfaces that fit the estimated corrective distortions $\tilde{\mathrm{p}}_i^{\mathrm{H}^j} - \tilde{\mathrm{p}}_i^j$ is given by

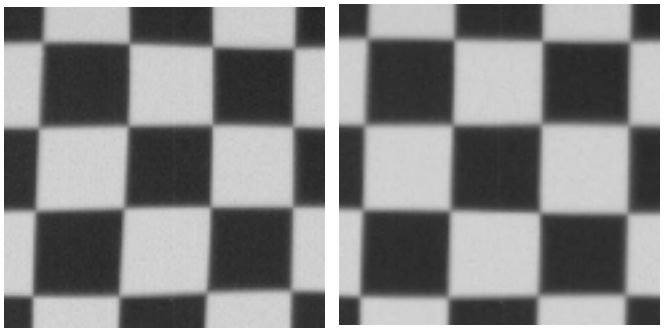
$$\rho = \min_{B_{Hz}^{k,l}, B_{Vz}^{k,l}} \sum_{i,j} \left\| \hat{\mathrm{p}}_i^{\mathrm{H}^j} - \tilde{\mathrm{p}}_i^j - \begin{bmatrix} \sum_{k,l} B_{Hz}^{k,l} * \mathcal{N}_u^k * \mathcal{M}_v^l \\ \sum_{k,l} B_{Vz}^{k,l} * \mathcal{N}_u^k * \mathcal{M}_v^l \\ 1 \end{bmatrix} \right\|^2 \quad (7)$$

- The distortion vectors are given by the 2 distortion B-spline surfaces. The remaining intrinsic & extrinsic parameters are initialised using Bouguet's solution on the currently undistorted pixel coordinates.

- The parameter vector to optimize is

$$\Phi_{5+2*R*S-8+6N} = [f_x, f_y, \alpha, u_0, v_0, B_{Hx}^{1,1}, ..., B_{Hx}^{R,S}, B_{Vx}^{1,1}, ..., B_{Vx}^{R,S}, t_x^1, t_y^1, t_z^1, \alpha^1, \beta^1, \gamma^1, ..., t_x^N, t_y^N, t_z^N, \alpha^N, \beta^N, \gamma^N]^T$$

Results

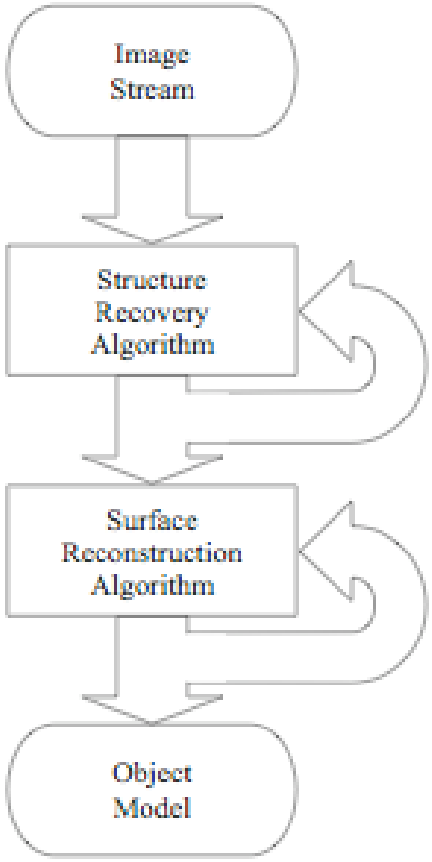


Distorted and corrected images.

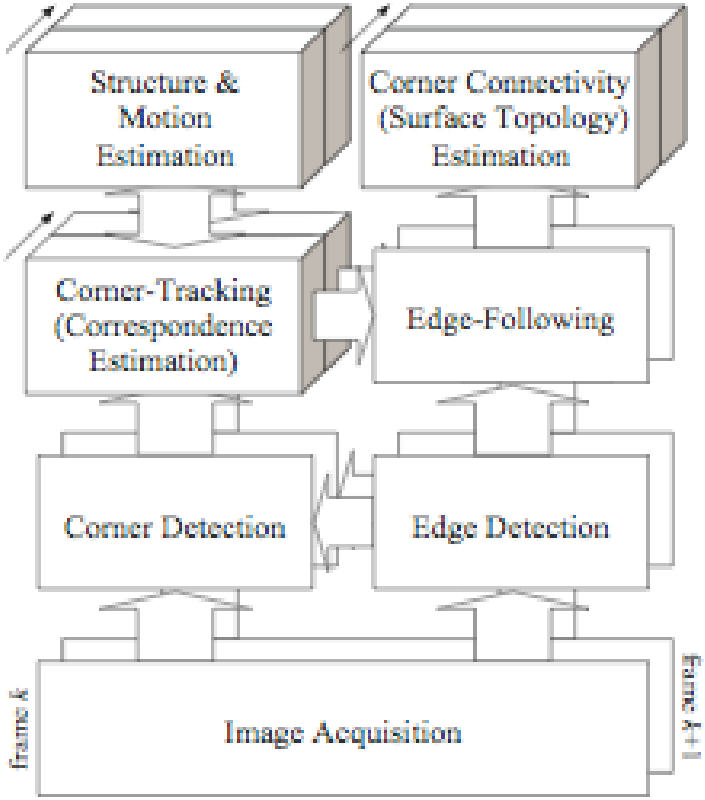
Re-constructing 3D object from an image sequence

- Involves correspondence estimation, structure & motion analysis, surface reconstruction
- Correspondence estimation (Redert, Hendricks & Biemond, 1999):
  - characterises the apparent motion of an imaged scene relative to the camera, either by the displacement of a discrete set of features or by the instantaneous velocities of brightness patterns
  - tracking a set of features: two-frame & long-sequence based
- Feature-based structure & motion estimation (Jebara, Azarbayejani & Pentland, 1999):
  - 2-frame; or iterative and recursive multi-frame
  - 1st, the rigid motion between the views is recovered; and 2nd, the motion estimate is used to recover the scene structure.
- Surface reconstruction:
  - Define surfaces to pass through recovered 3D feature points

3D metric object modelling from uncalibrated image sequences (Cadman & Tjahjadi, IEEE Trans SMC, 2004)



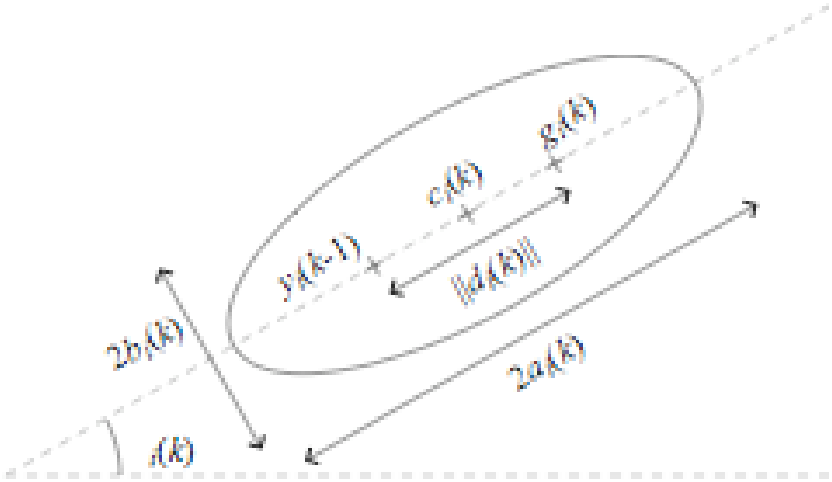
System architecture



Interactions between system modules

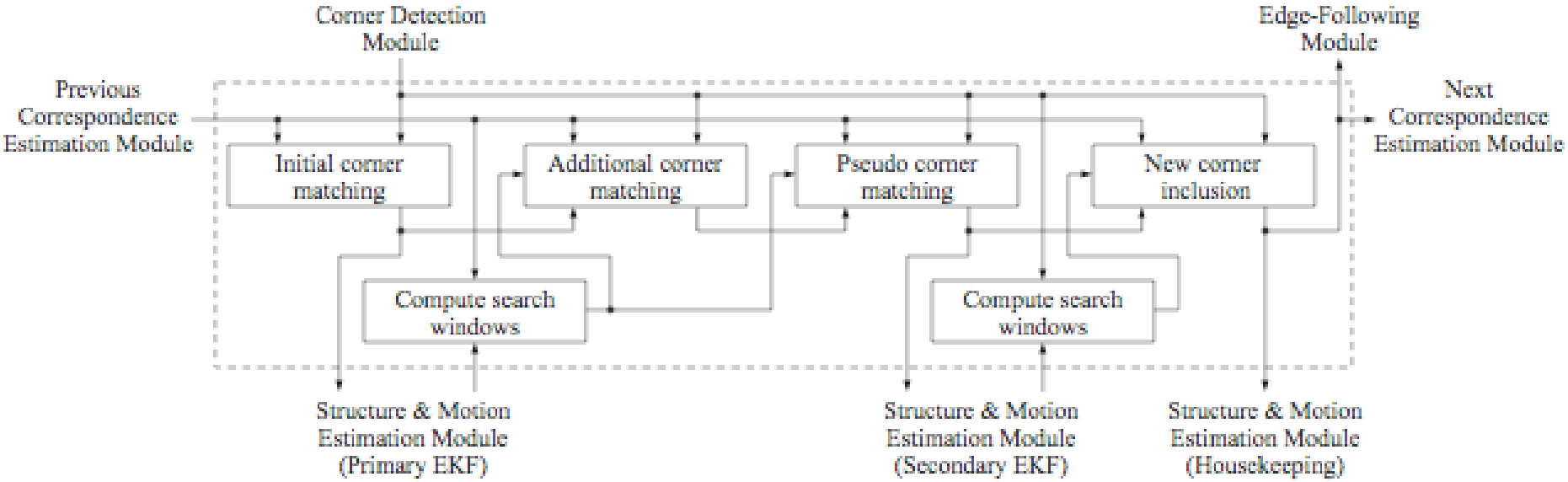
General matching procedure

- Unguided matching:
  - when the transformation between the two most recent camera positions (i.e., viewpoints) is unknown
  - given a feature point in the previous image, a circular area centred at the same location in the current image is searched for the feature's corresponding point
- Guided matching:
  - when the motion of the camera between its two most recent viewpoints has been estimated
  - uses a dynamic search window



Dynamic search window

Corner tracking



- **Initial corner matching:** unguided matching to give ≥ 8 matched points
- **Additional corner matching:** expected image locations of unmatched feature points are used to guide matching
- **Pseudo corner matching and expiration:** to handle disappearance, occlusion & unsuccessful detection of corners
- **Inclusion of new corners:** guided matching on original sets of feature points and all expired pseudo corners using more lenient thresholds

Structure recovery

- To estimate the motion of the camera and structure of a scene from the point correspondences established, as well the camera intrinsic parameters
- The relationship between a 3D point $p_i(k-1)$ that moves to $p_i(k)$ due to the rigid motion relative to the camera between two consecutive views is

$$p_i(k) = \delta R(k-1)p_i(k-1) + \delta t(k-1) \tag{8}$$

  - where *i* is the feature point index; *k* is the image frame index; and $\delta R(k)$ and $\delta t(k)$ are the interframe rotation and translation respectively.
- Given $N_p(k-1)$ point correspondences $(p_i(k-1), p_i(l))$, $\delta R(k-1)$ and $\delta t(k-1)$ are obtained as a solution to

$$\underset{\delta R(k-1) \text{ and } \delta t(k-1)}{\text{minimise w.r.t.}} \left\{ \sum_{i=0}^{N_P(k-1)-1} \|p_i(k) - (\delta R(k-1)p_i(k-1) + \delta t(k-1))\|^2 \right\} \tag{9}$$

  - the measurement model comprising: camera model, structure model, motion model;
  - the solution is provided by extended Kalman filter

- Camera model

$$\text{pers}(p_i(k)) = \begin{pmatrix} y_{i,u}(k) \\ y_{i,v}(k) \end{pmatrix} = \begin{pmatrix} p_{i,x}(k)/(1+p_{i,z}(k)\beta) \\ p_{i,y}(k)/(1+p_{i,z}(k)\beta) \end{pmatrix} \qquad (10)$$

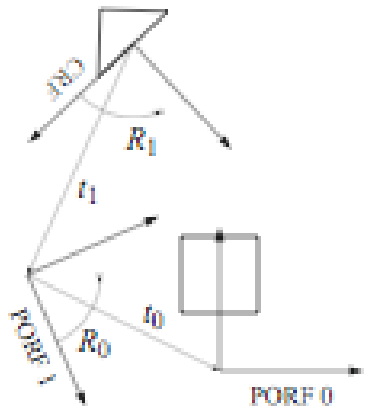
  - where $y_i(k) = (y_{i,u}(k), y_{i,v}(k))^{\mathrm{T}}$ corresponds to the projection of $p_i(k) = (p_{i,x}(k), p_{i,y}(k), p_{i,z}(k))^{\mathrm{T}}$, β is reciprocal of camera focal length

- Structure model: the 3D location of feature point $p_i(k)$ is

$$\begin{pmatrix} p_{i,x}(k) \\ p_{i,y}(k) \\ p_{i,z}(k) \end{pmatrix} = \begin{pmatrix} y_{i,u}(k) \\ y_{i,v}(k) \\ 0 \end{pmatrix} + \alpha_i \begin{pmatrix} y_{i,u}(k)\beta \\ y_{i,v}(k)\beta \\ 1 \end{pmatrix} \qquad (11)$$

  - where α_i is depth of feature point

- Motion model:
  - the position and orientation of the object-centred coordinate system is permitted to vary with time, resulting in pseudo object reference frames (PORFs)



- Fig. Relationship between 2 PORFs and the current camera reference frame (CRF). Triangle - camera viewing pyramid; square - scene.

Surface reconstruction

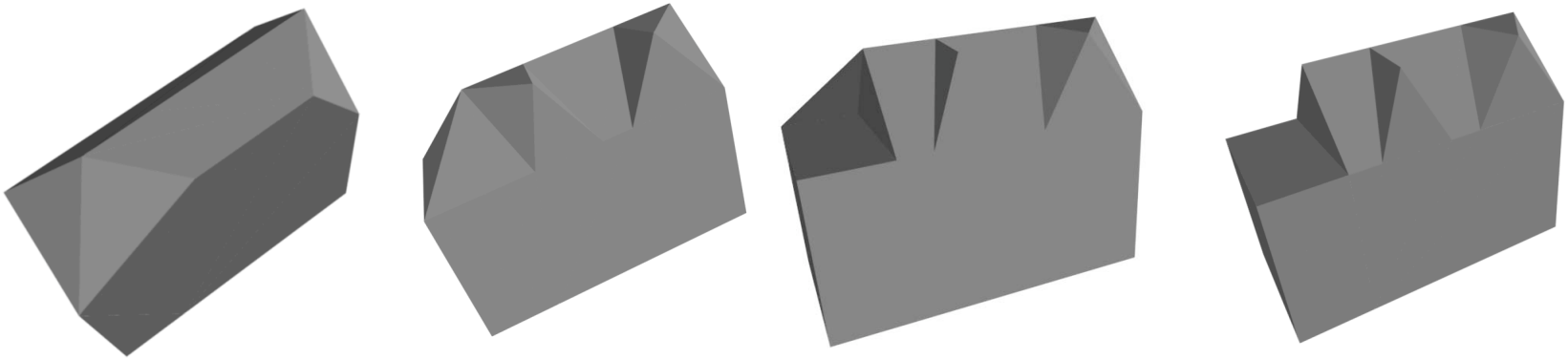
- A recursive framework which incorporates a recursive score (indicating the likelihood of any 2 feature points begin adjacent to one another on the true surface) is used to estimate the complete topology of the imaged scene.
- Corner connectivity

$$\mathrm{conn}_{i,n}(k) = \mathrm{conn}_{i,n}(k-1) - \frac{\mathrm{conn}_{i,n}(k-1) - \mathrm{dir}(y_i(k), y_n(k))}{\lambda_{i,n}(k)} \quad (12)$$

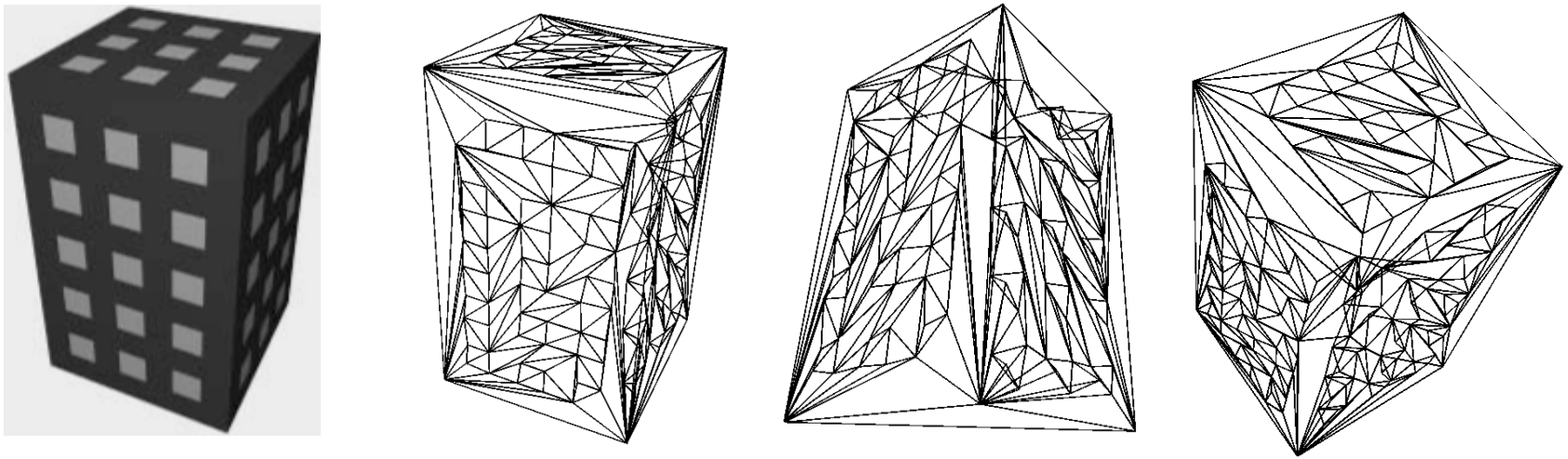
  - where $\mathrm{dir}(y_i(k), y_n(k))$ is a directness score and $\lambda_{i,n}(k)$ denotes the number of frames in which both features have appeared together
  - If greater than threshold $\Rightarrow$ a potential edge
- Constrained triangulation
  - Beginning with a convex hull enclosing the object feature points, the triangulated surface mesh is iteratively refined to ultimately yield a constrained triangulation that can produce views consistent with the original images of the scene.
  - Visibility constraint: none of the surface's visible facets should obscure or be occluded by a potential edge.

Example results

- Evolution of the surface overlaying recovered structure of a simulated object

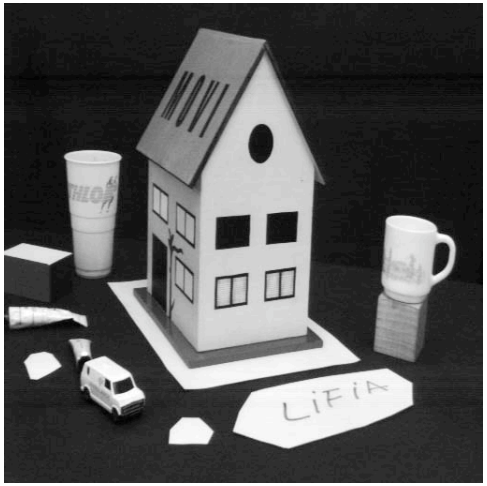


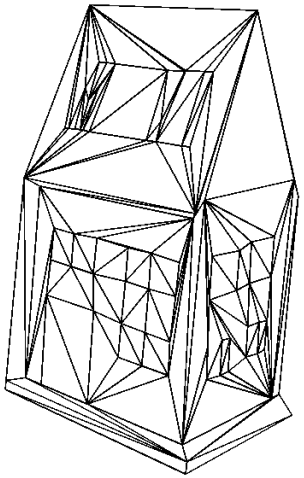
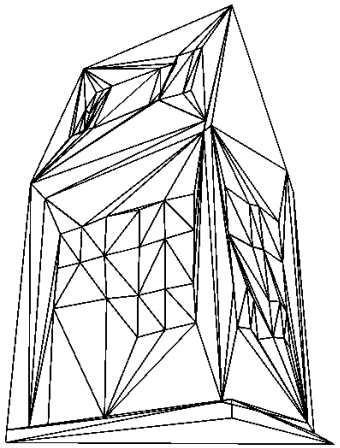
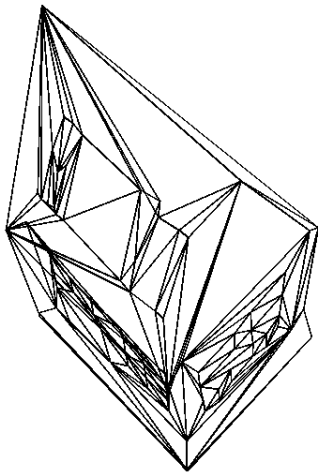
- Recovering an object model from a raytraced box sequence



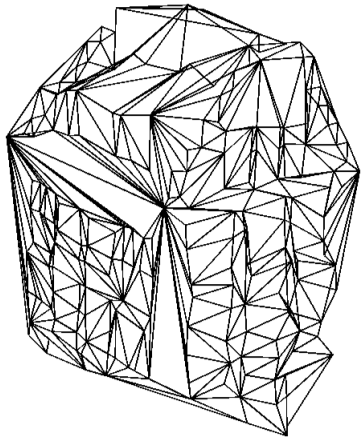
Example results

- Recovering an object model from the MOVI sequence



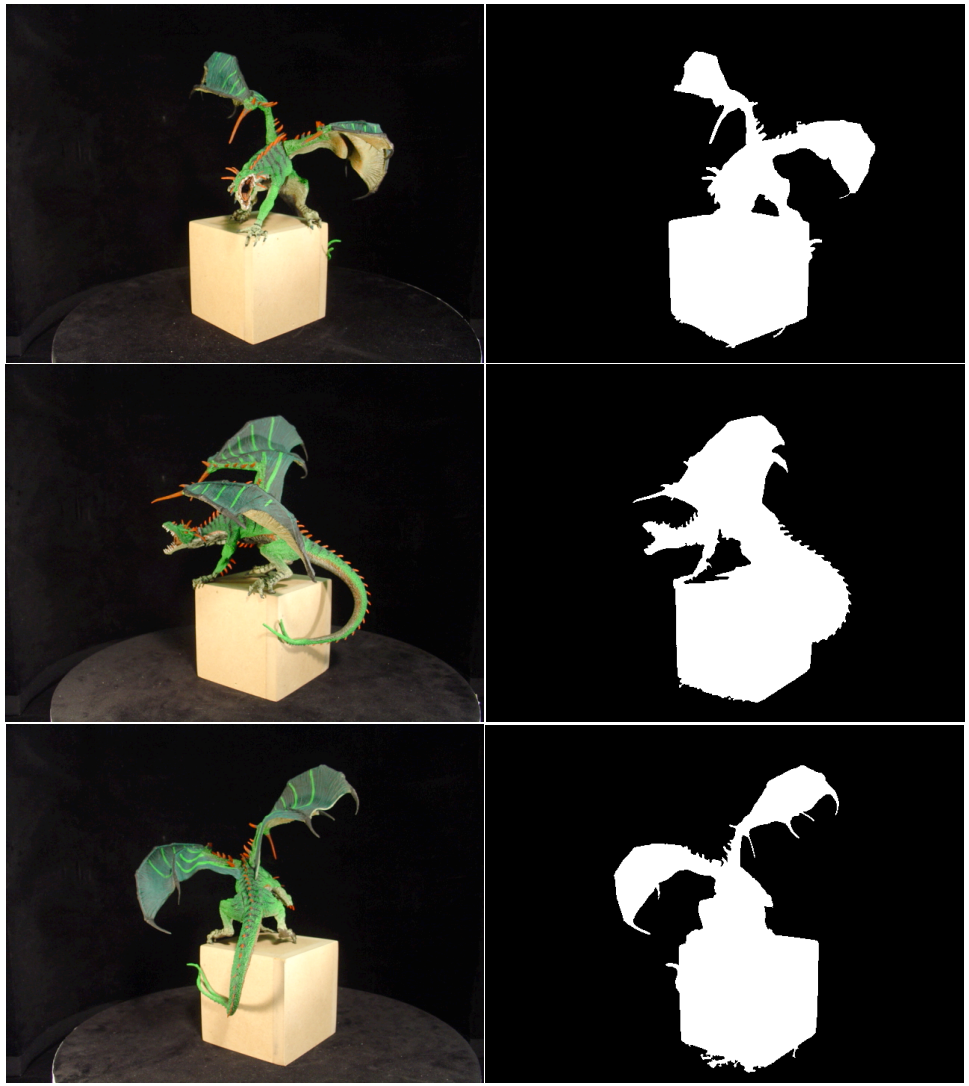




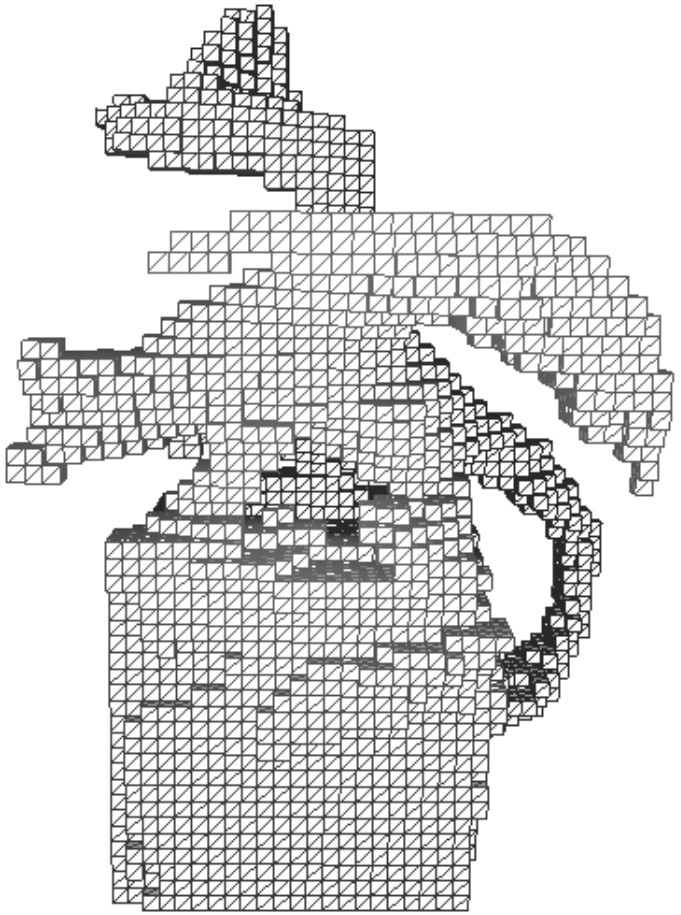
- Recovering an object model from the hotel sequence

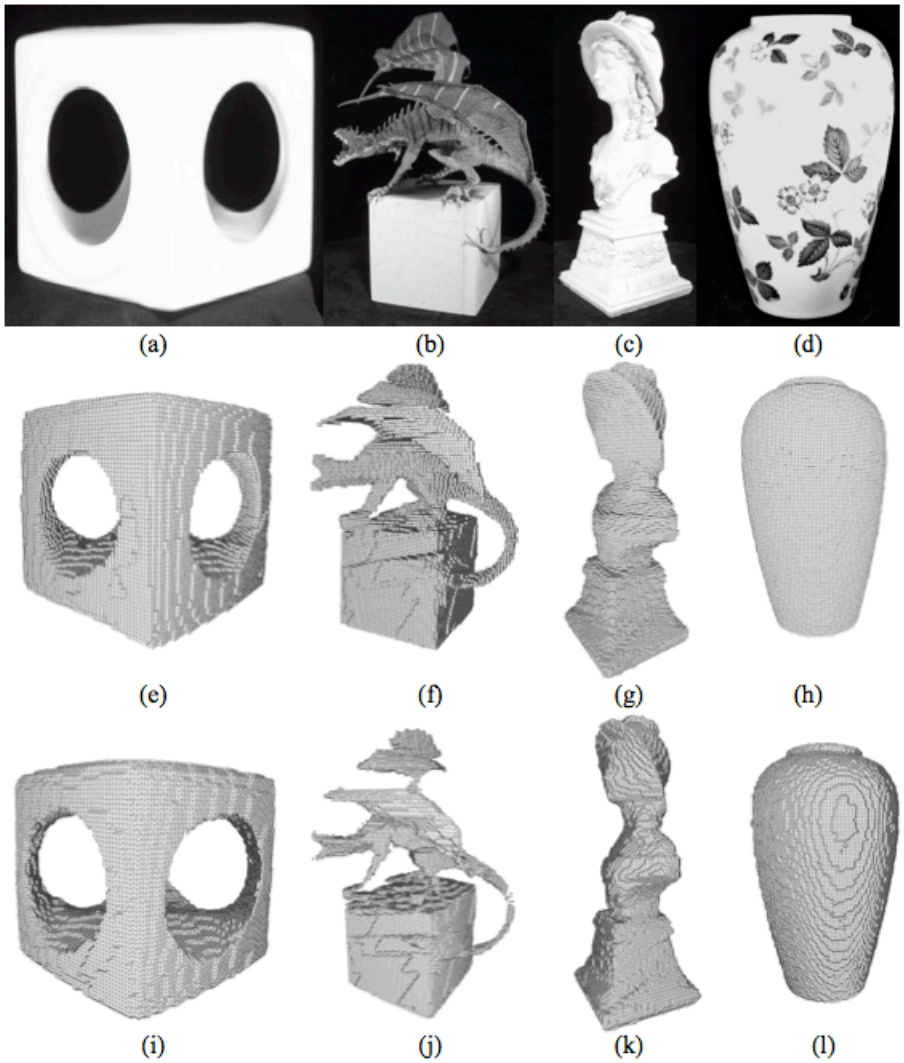
Shape from silhouette (SfS) techniques

- **3D visual hull:** intersection of multiple 3D cones that are created by backprojection of 2D silhouettes of different views of an object onto 3D space.
- **Octree:** constructed by projecting an initial bounding cube (which encloses an object in 3D space) onto multiple images of the object taken at different views, and splitting the cube into eight octants if the projection intersects a silhouette
- **Octants:** outside, inside or intersection
- **Marching Cube (MC)** (Lorensen & Cline, 1987):
  - estimates surface triangles from intersection octants, and the location of the triangles are determined by the configuration of inside vertices of an intersection octant
  - MC generated surface may contain unexpected holes or discontinuities due to: connectivity of octants, ambuiguity of MC algorithm, errorneous camera calibration, and imperfect silhouette images.

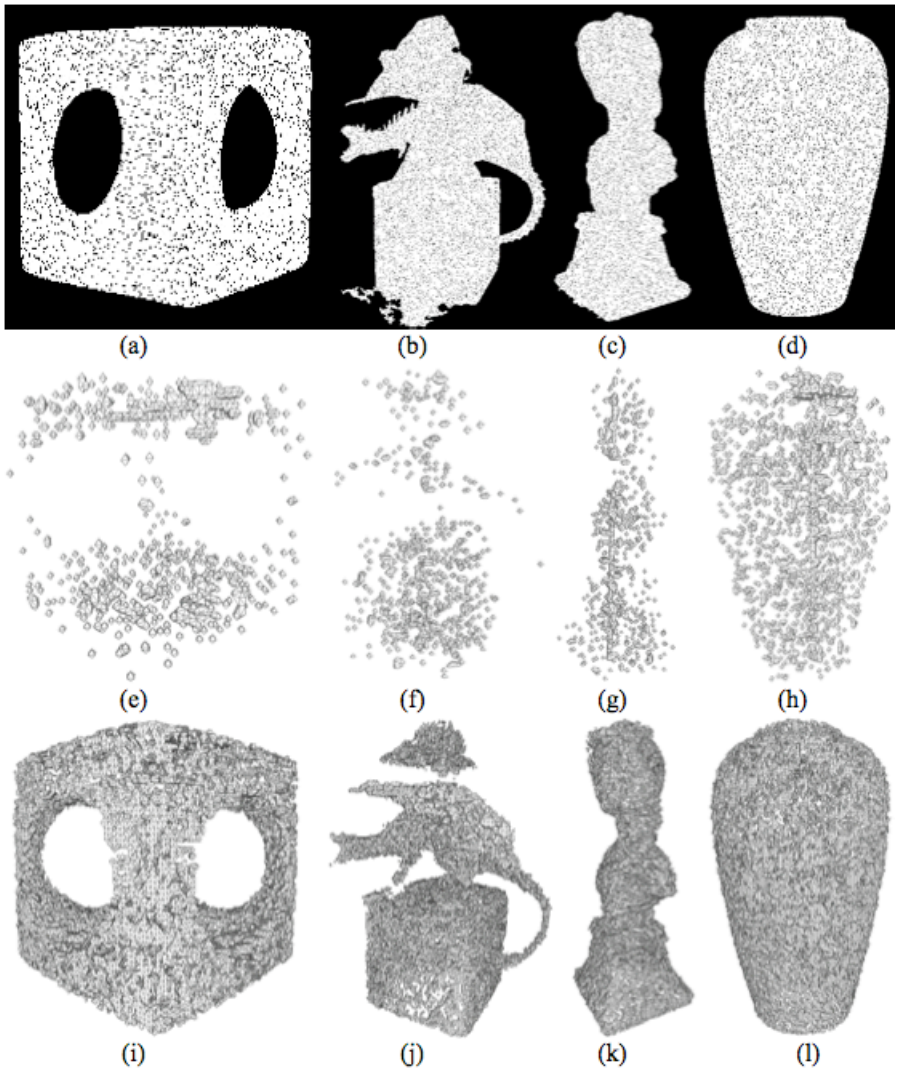




Images **Silhouettes** **Octree**



8-level octrees and MC surfaces of four test objects: (a)-(d) Images; (e)-(h) Octrees. ; (i)-(l) MC surfaces.

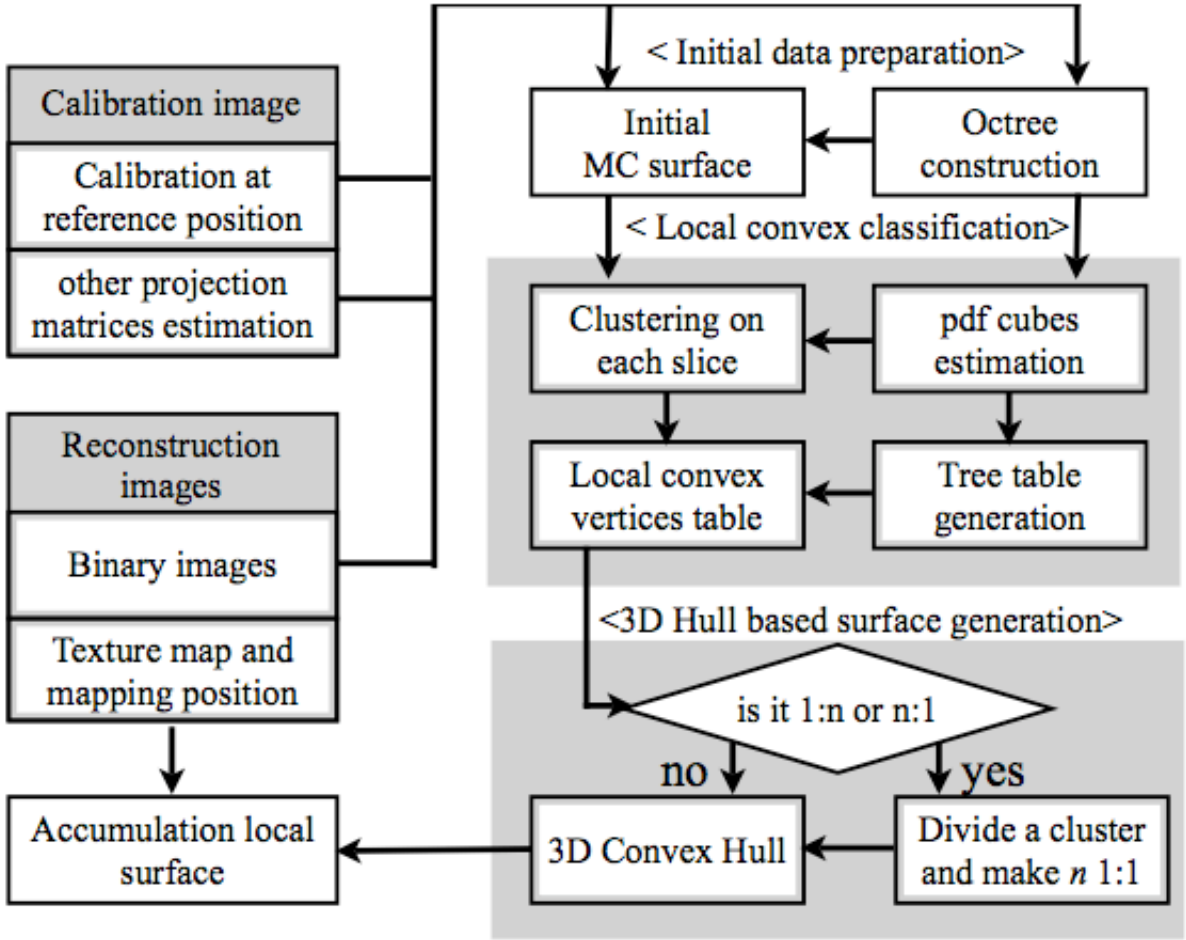


(a)-(d) Silhouettes with 10% noise; (e)-(f) MC surfaces from silhouette with 5% noise; (i)-(l)VMC results from silhouette with 10% noise.

Surface reconstruction

- MC
  - assumes that intersection octants may include an actual surface which crosses an edge joining two vertices of a surface octant with opposite status, i.e., inside and outside
  - erroneous decision on an inside vertex supersedes other statuses previously defined in other silhouettes ⇒ lost of surface patches
- Voting MC (VMC) (Yemez & Schmitt, 2004)
  - counts the number of cases classified as outside and identifies an outside vertex if the vote is greater than a threshold
- Delaunay Triangulation (DT) (Aurenhammer, 1991)
  - constructs a convex surface by defining tetrahedrons from 3D points.
  - characterises each tetrahedron by not allowing any point within its circumsphere

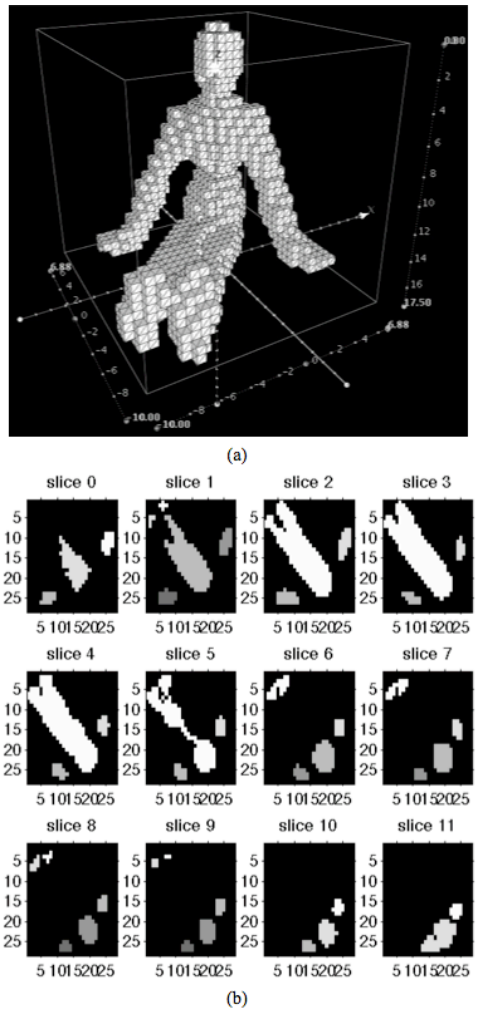
Local hull-based surface reconstruction (Shin & Tjahjadi, IEEE Trans IP, 2008)



Overview of surface construction process

3D objects & volumetric data slicing

- 3D object properties
  - Connectivity: surface of an object should cover the object tightly without any unattached object segments
  - Continuity: a shape of local convexity is similar to adjacent convexity if they are connected ⇒ object needs to be sliced infinitesimally
- Uses best VMC vertices
- Uses octree vertices to define a local convexity
- Quantized octree slice S_i^{ocq}
  - every four points are from the same octant
  - a binary image plane where a nonzero point represents an octant
  - of a non-convex object can have multiple clusters that are linked 8-neighbouring points
- Quantized MC slice S_i^{mcq}: decision on clustering and connecting of clusters is based on a Bayesian rule and a priori information from the quantised octree slice



(a) Octree; (b) octree slices

Identifying a local convexity

- A local convexity is identified by: clustering on S_i^{mcq} and connecting clusters between slices. Given a cluster conditional pdf $p(\vec{t}\,|c_i)$, the problem of clustering is solved by searching for the maximum probability. ($\vec{t}$ -test data, c_i -class *i*)
- Using Parzen density estimator

$$p_i(\vec{t}|c_j) = \frac{1}{n_j}\sum_{i=0}^{n_j} w(\vec{t} - \vec{t}_i) \qquad (13)$$

n_j - number of data in cluster *j*

- Cluster decision function

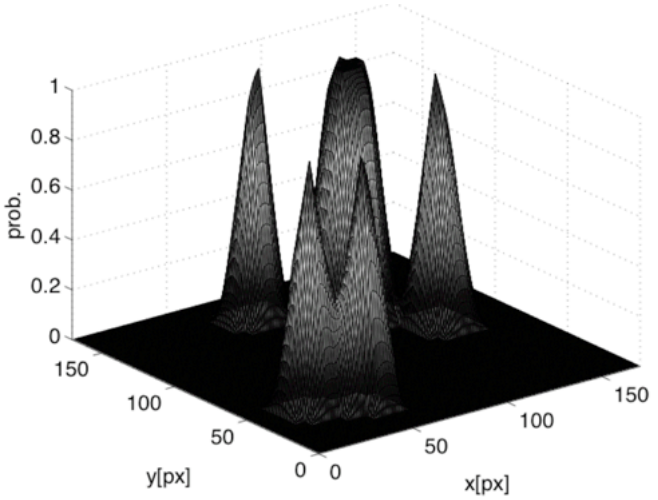
$$d(\vec{t}) = \max_j \{p_i(\vec{t}|c_j)P_i(c_j) \qquad (14)$$

- Decision function for connecting a cluster

$$e(c_{id}, i) = \max_j\{\sum_{\vec{t}\in c_{id}} p_{i+1}(\vec{t}|c_j)P_{i+1}(c_j)\} \qquad (15)$$

- Correlation coefficient between CID and TID's:

$$g(c_m, c_n) = \left|\frac{(\sum_i p_j(\vec{t}_i|c_m)p_{j+1}(\vec{t}_i|c_n))^2}{\sum_i p_{j+1}^2(\vec{t}_i|c_n)\sum_i p_j^2(\vec{t}_i|c_m)}\right| \qquad (16)$$



(a)

SNO	CID	HID	TID	Cr
0	0	x	6	0.9615
	1	x	5	0.6989
	2	x	7	0.9088
1	3	x	8	0.0304
	4	x	8	0.0276
	5	1	8	0.7882
	6	0	9	0.9716
	7	2	10	0.9357
2	8	5, 3, 4	11	0.9762
	9	6	12	0.6858
	10	7	13	0.5963
3	11	8	14	0.9872
	12	9	15	0.6700
	13	10	16	0.7869

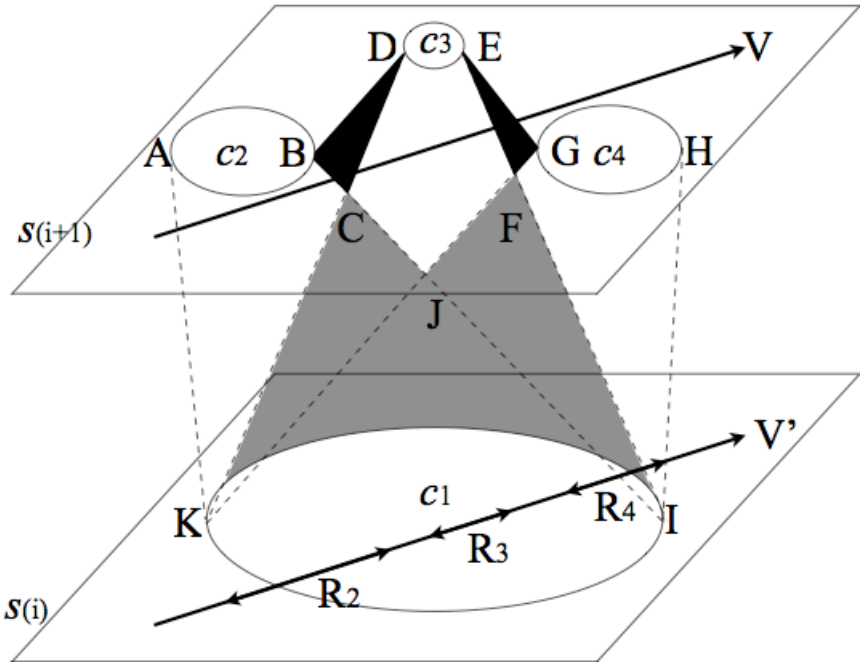
SNO	CID	HID	TID	Cr
4	14	11	17	0.7323
	15	12	18	0.9465
	16	13	19	0.8629
5	17	14	22, 20	0.2870
	18	15	21	0.9830
	19	16	23	0.9278
6	20	17	24	0.9510
	21	18	25	0.9212
	22	17	26	0.9634
	23	19	27	0.9189
7	24	20	28, 29	0.4622
	25	21	30	0.8645
	26	22	31	0.9621
	27	23	32	0.9805

(b)

(a) 3D pdf cube containing every cluster conditional pdf in a quantised MC slice.
(b) Part of a tree table from slice 4 to 7.

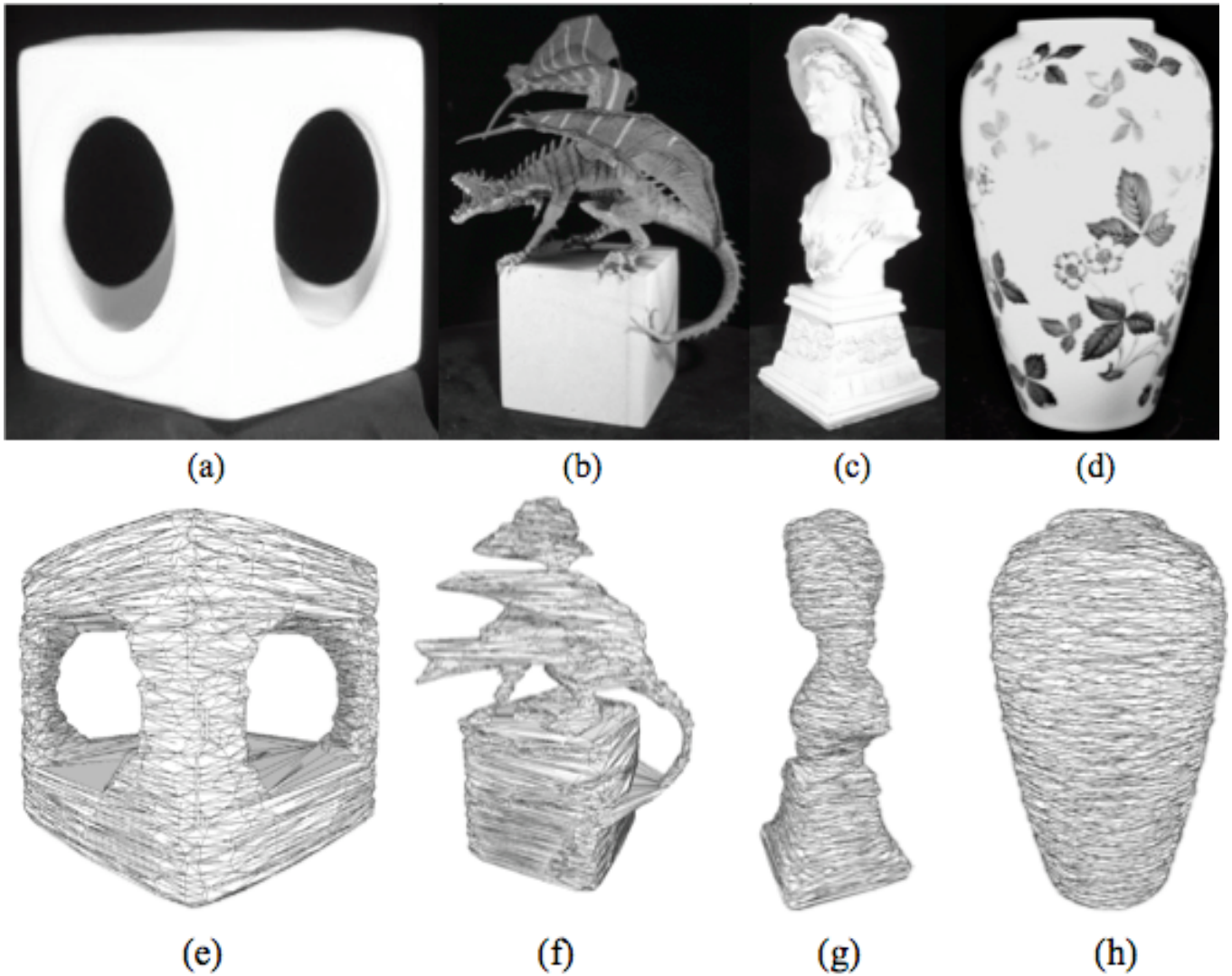
Local surface construction

- A local convexity is defined by two connected clusters in different slices
- If an object is not convex, a local convexity can have multiple connections
- Divide multiple connections into *n* 1:1 connections with an appropriate division so as to minimise possible duplication of surface patches in the common area
- The division is done along the best representative vector of the multiple clusters (determined using eigen analysis) and according to the normalised data distribution along this vector.



1:n branching case. To avoid smoothing, the cluster C_1 is divided into 3 subregions, R_2, R_3 and R_4 on the projection of the eigen vector V' and *n* 1:1 connections are made.

Results



(a)-(d) test objects: burner, dragon, bust, vase. (e)-(f) the reconstructed surfaces.

Thank you for your attention.

Any questions?

THE UNIVERSITY OF
WARWICK