Computer Vision: 10b. Hybrid Modeling

I-Chen Lin, Assistant Professor
Dept. of CS, National Chiao Tung University

Introduction

- Represent an environment.
  - Geometry-based
    - Arbitrary view points.
    - Conventional graphics techniques are applicable. (ex. rendering, shading, etc.)
    - Modeling details is not easy.
  - Image-based
    - Limited view points.
    - Limited lighting conditions.
    - Stereo: find correspondences (not always robust)
    - IBR: Avoid modeling details.

Background

- Structure from motion
  - Sensitive to noise in image measurements.
  - No-linear optimization: local minima problem.

- Stereo correspondence
  - Successful only when the images are similar in appearance.

- Image-based rendering
  - Requiring dense samples
  - Infeasible for large-scale environments.

Introduction

- How about taking the advantages of both approaches?
  - A hybrid approach.
    - Geometry + Image - based

Ref:

• Image-based Modeling and Rendering, SIGGRAPH’99 course notes.

• P. E. Debevec, C.J. Taylor, J. Malik, "Modeling and Rendering Architecture from Photographs: A Hybrid Geometry- and Image-Based Approach", Proc.SIGGRAPH'96, pp. 11-20.

Modeling and Rendering Architecture from Photographs:

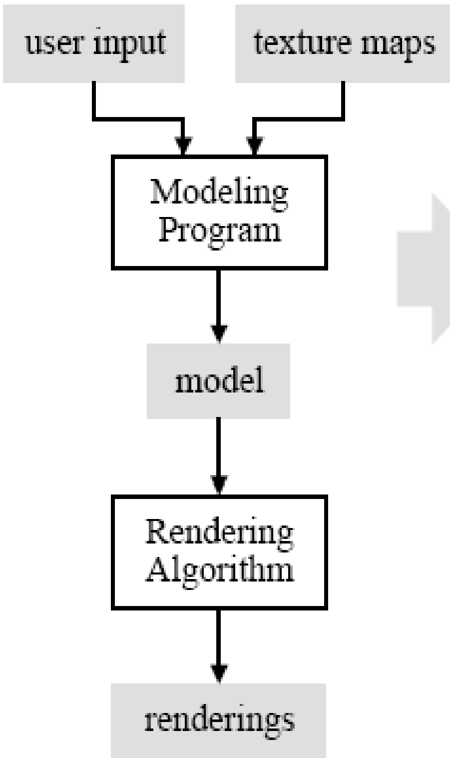
A Hybrid Geometry- and Image-Based Approach

Proposed by:

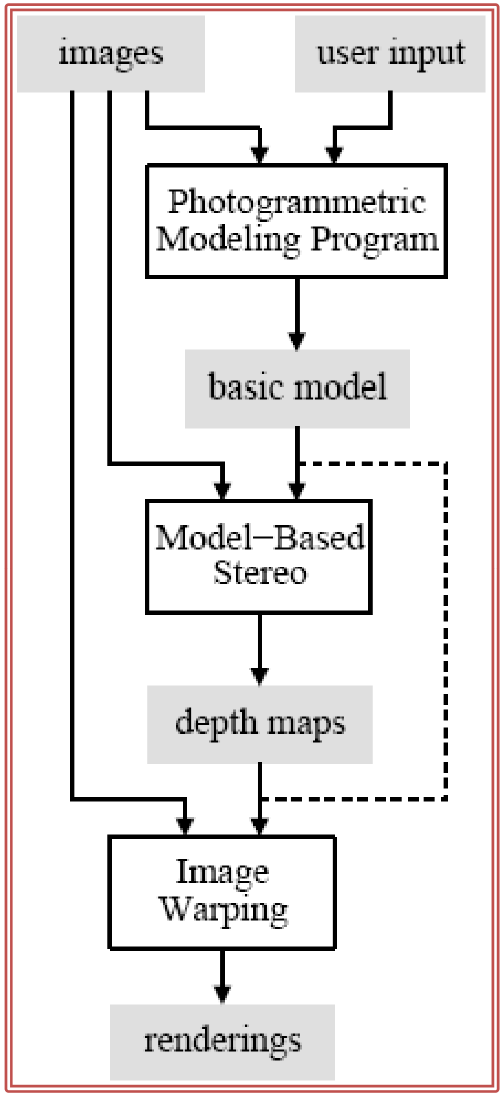
P. E. Debevec, C.J. Taylor, J. Malik

Proc. SIGGRAPH'96

Introduction



Geometry-based



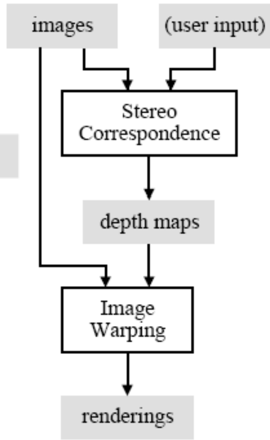
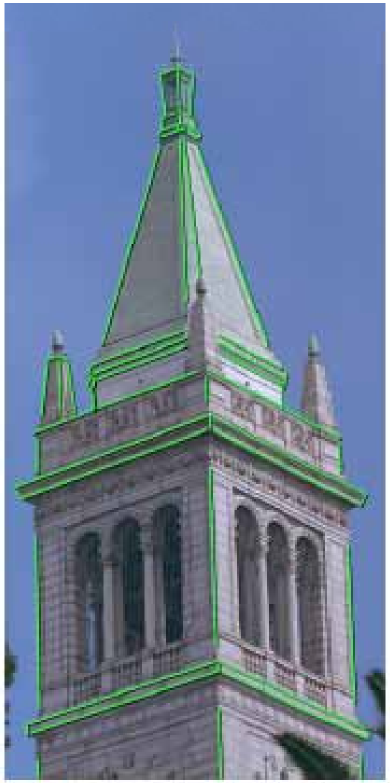


Image-based

Overview

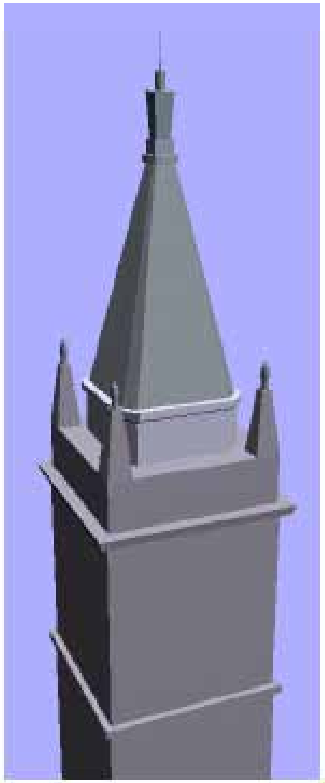
- Geometry-based + Image-based
  - Photogrammetric modeling
    - Recasting the structure.
    - Modeling based on a constrained hierachy of parametric primitives.
  - View-dependent texture mapping
  - Model-based stereo
    - Refining the initial approximate model.

Photogrammetric modeling

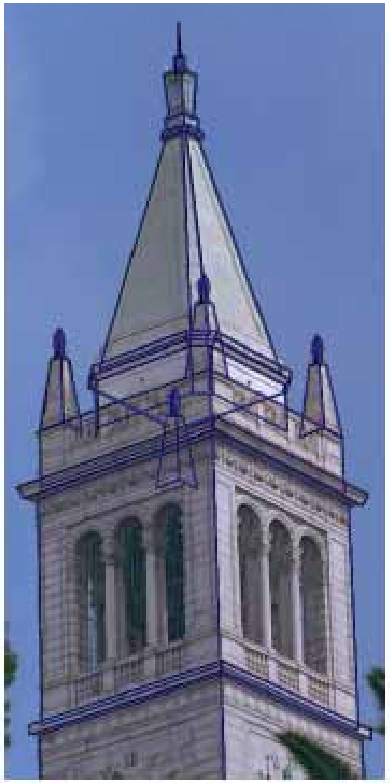


Façade:

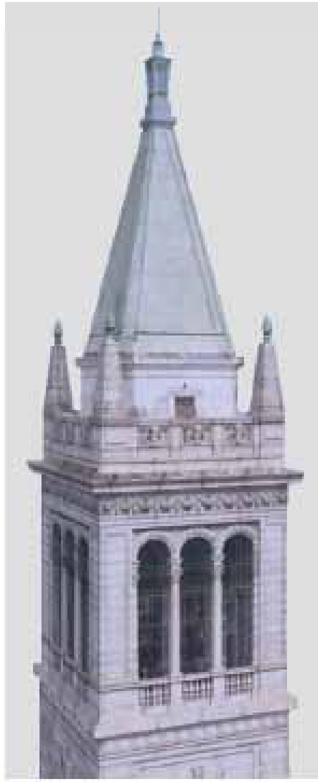
an interactive modeling system



Modeling based on primitives



Reprojection of the model



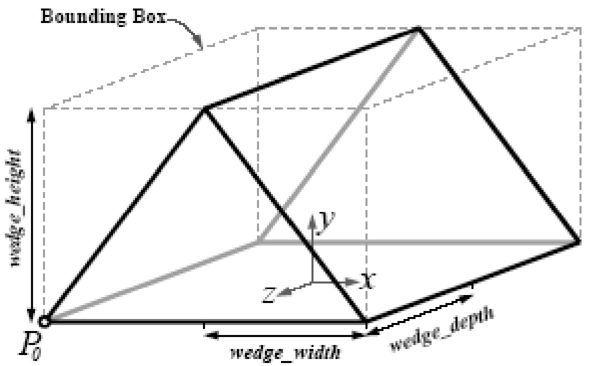
A synthetic view with view-dependent textures

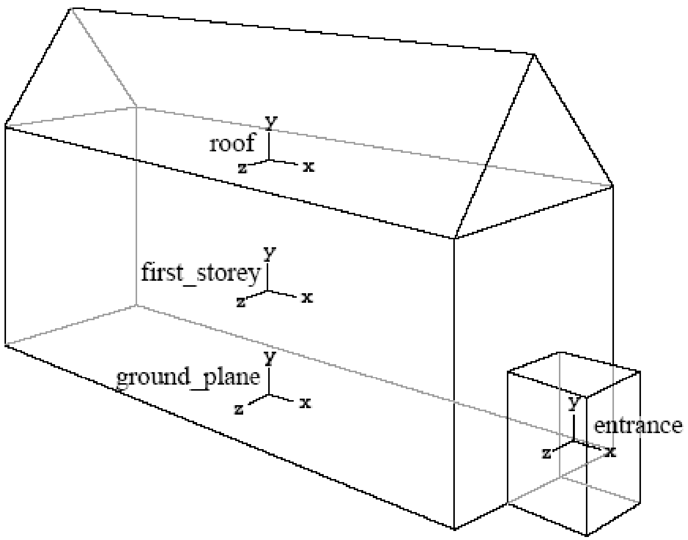
Photogrammetric modeling (cont.)

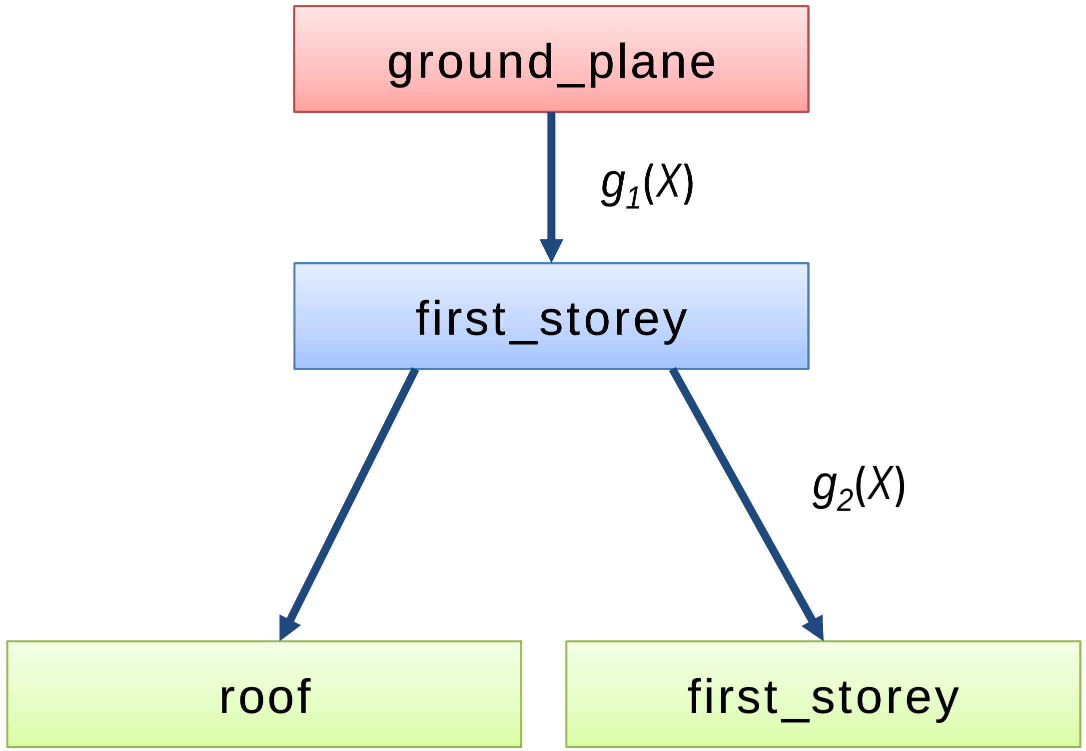
- User assistance
  - Marking edges in images.
  - Corresponding the edges in images to the edges
    - Using "intelligent scissors" for sub-pixel accurac
  - Constraining the size, positions, etc.
    - E.g. equal length and width, symmetric constrai

Model representation

- A constrained hierarchical model of primitives (blocks).







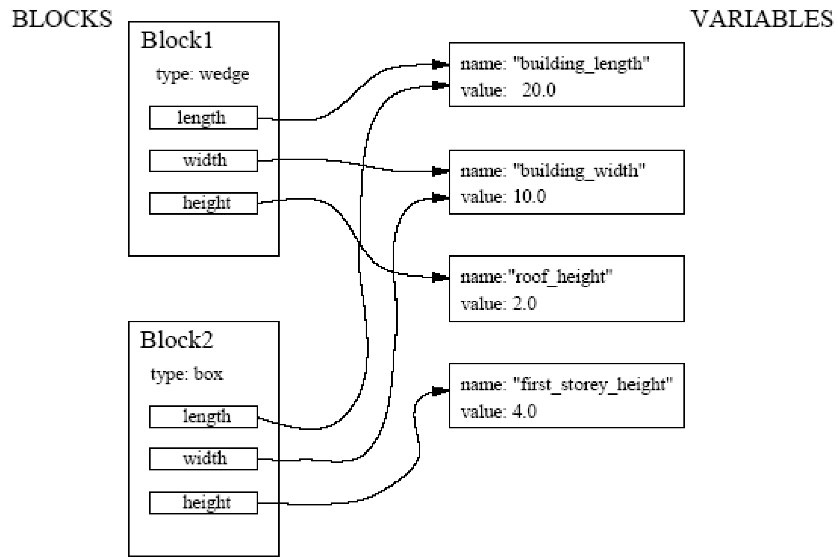
Model representation (cont.)

- $g_i(X)$: a relation between blocks
  - *R* and *t*

$$P_w(X) = g_1(X)\ldots\ldots g_n(X)P(X)$$

$$v_w(X) = g_1(X)\ldots\ldots g_n(X)v(X)$$

  - Using constraints to reduce the unknown parameters (from thousands to dozens).



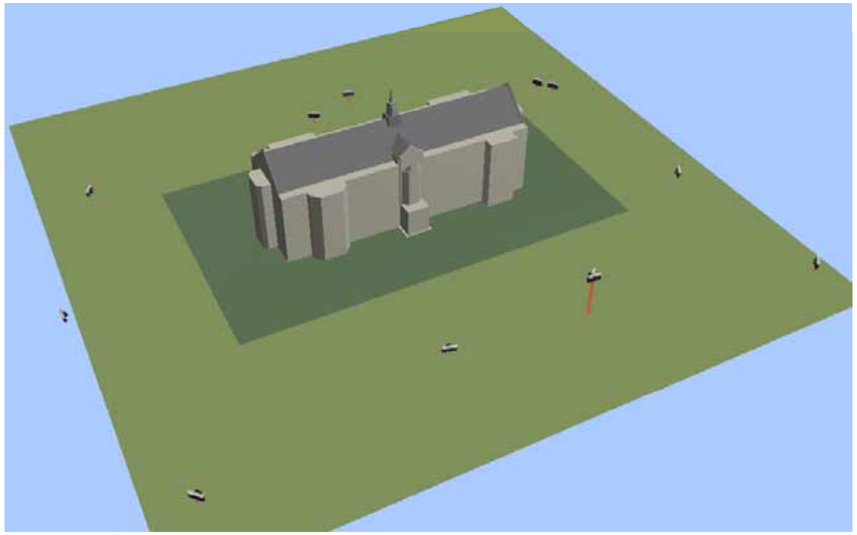
Reconstruction

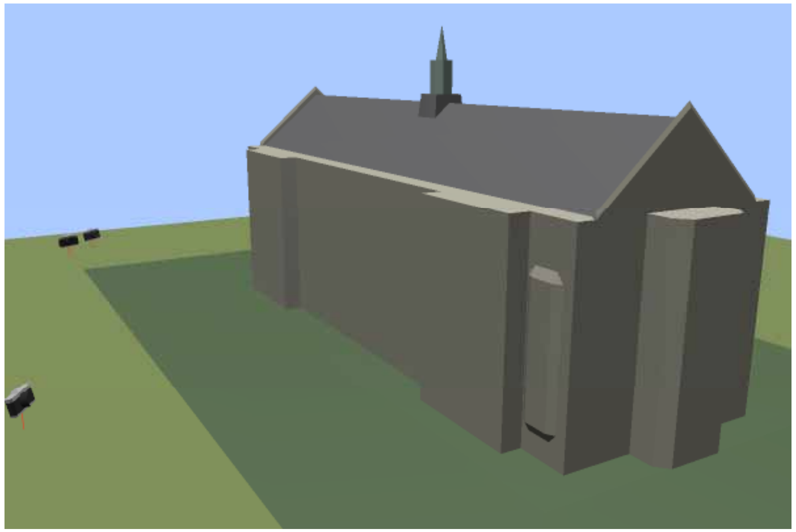
- Goal: to fit marked edges of the model to those on images.
- A calibrated camera (with known *f*, radial distortion, etc.)

Reconstruction (cont.)

- Unknown: model parameters, camera positions and orientations.

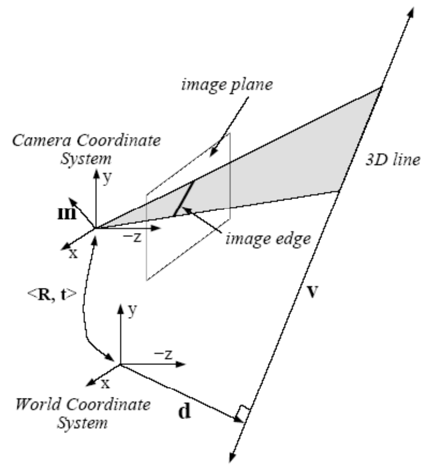
- Reconstruction steps
  - Initial estimate of the unknown parameters.
  - Further refinement.





Transformation

- R_j, t_j
  - Transformation between the camera coordinate system and the world coordinate system.
- <v, d>
  - A straight line in the W coord.



$$m = R_j\left(v \times \left(d - t_j\right)\right)$$

Initial Estimate

- Overview
  - First, estimating the R_j.
  - Second, estimating the t_j and model parameters.

- Estimating R_j:
  - Given an observed edge:

$$m' = \begin{pmatrix} x_1 \\ y_1 \\ -f \end{pmatrix} \times \begin{pmatrix} x_2 \\ y_2 \\ -f \end{pmatrix}$$

  - Since $m^T R_j v = 0$ (ideally),

$$\min O_1 = \sum_i \left(m^T R_j v_i \right)^2$$

Initial Estimate (cont.)

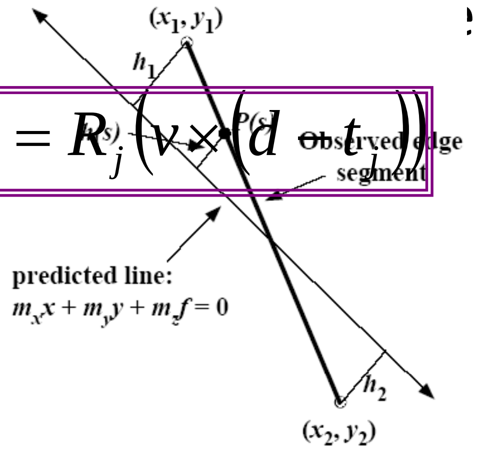
- Estimating t_j and model parameters:
  - Since $m^T R_j(d - t_j) = 0$ (ideally),

$$\min O_2 = \sum_i \left(m^T R_j \left(P_i(X) - t_j\right)\right)^2 + \left(m^T R_j \left(Q_i(X) - t_j\right)\right)^2$$

Estimating parameters

- Given R_j, t_j, and <v,d>, we can project t… e image plane.

$$m = R_j\left(v \times \left(d - t_j\right)\right)$$



- Minimize the error between the obser… predicted ones.

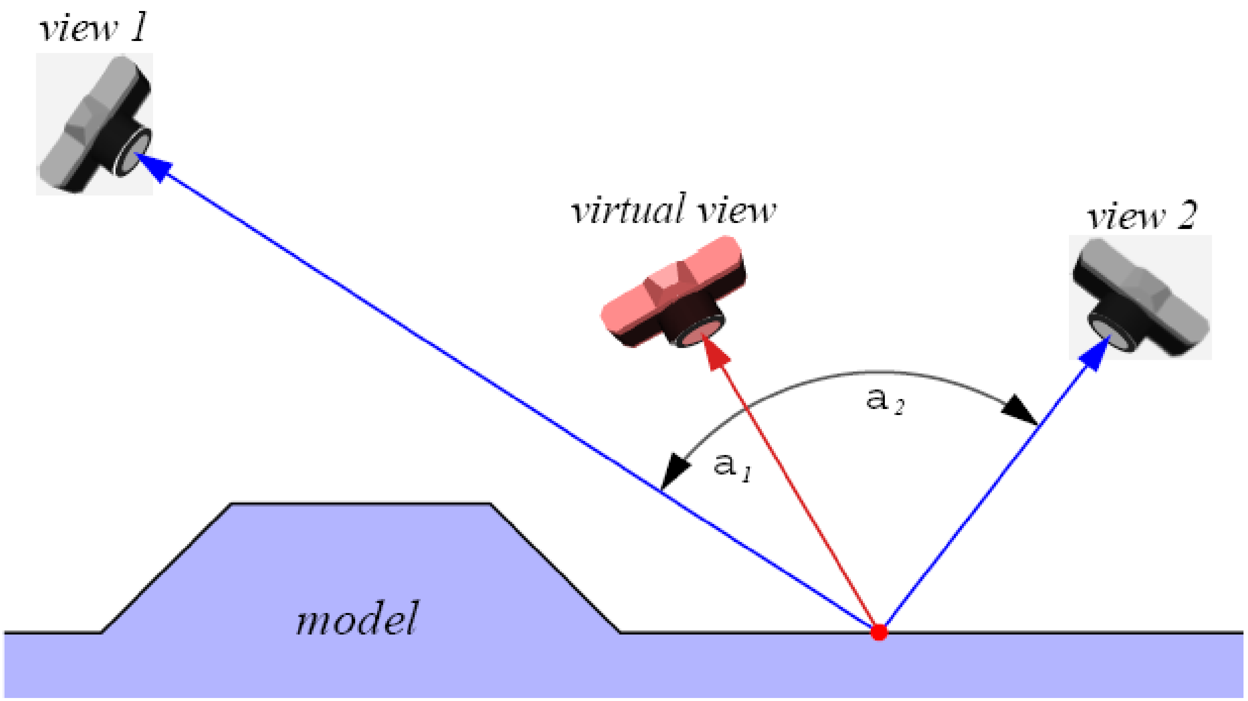
$$Err_i = \int_0^l h^2(s)ds = \frac{l}{3}\left(h_1^2 + h_1 h_2 + h_2^2\right)$$
$$= m^T\left(A^T B A\right)m$$

$$m = \begin{pmatrix} m_x & m_y & m_z \end{pmatrix}^T$$

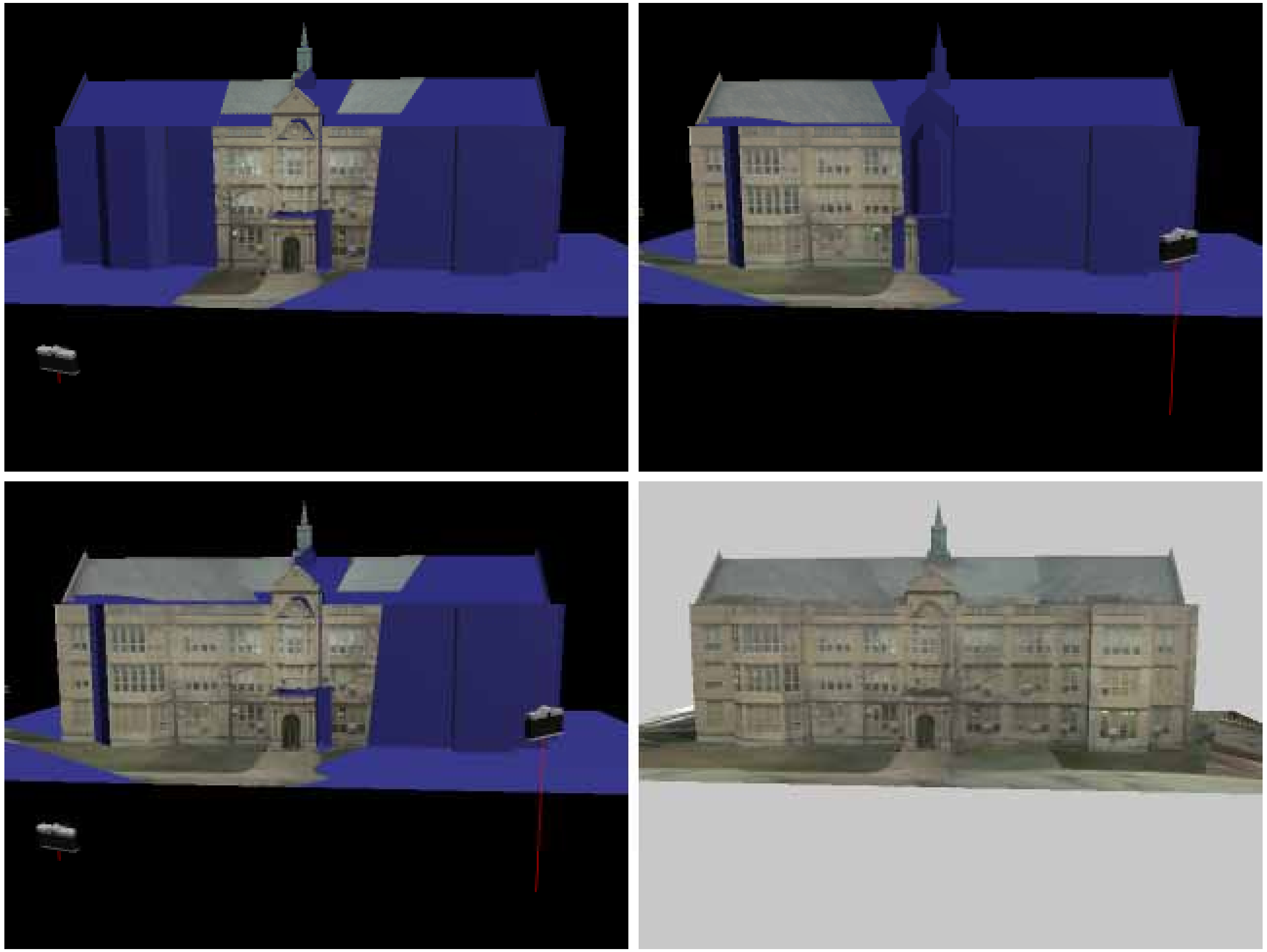
$$A = \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{pmatrix}$$

$$B = \frac{l}{3\left(m_x^2 + m_y^2\right)}\begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$$

View-dependent texture-mapping

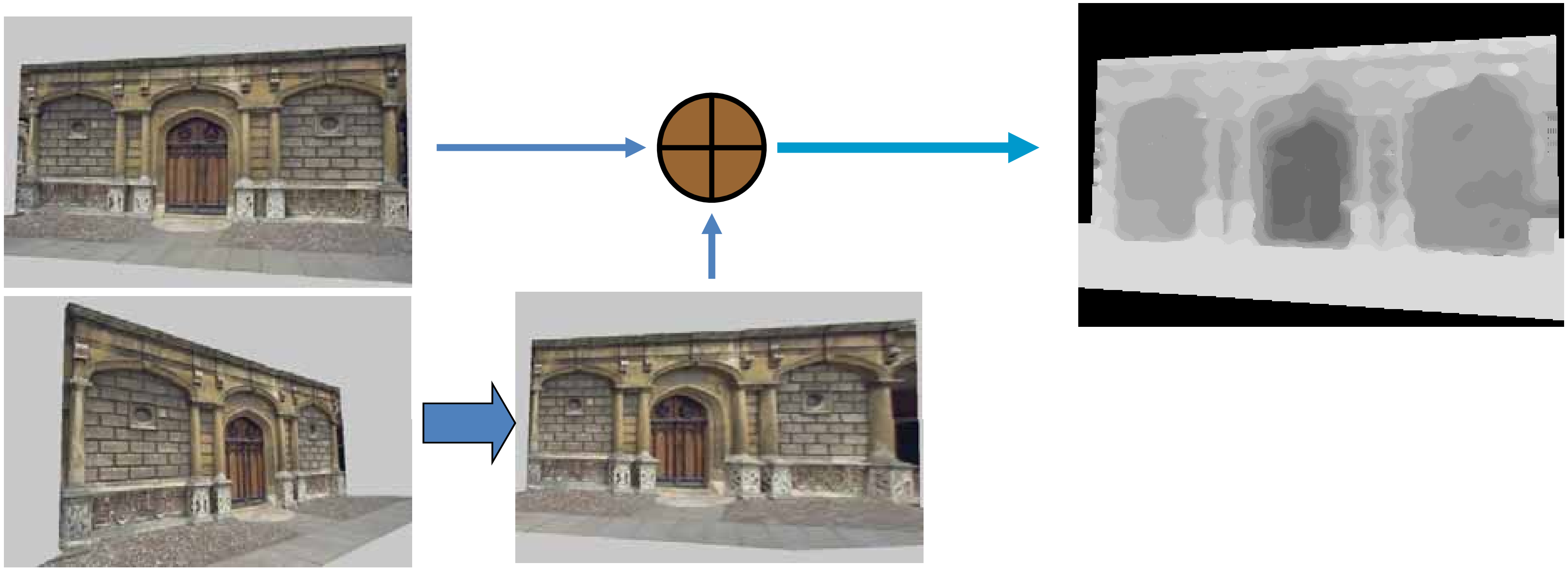


View-dependent texture-mapping

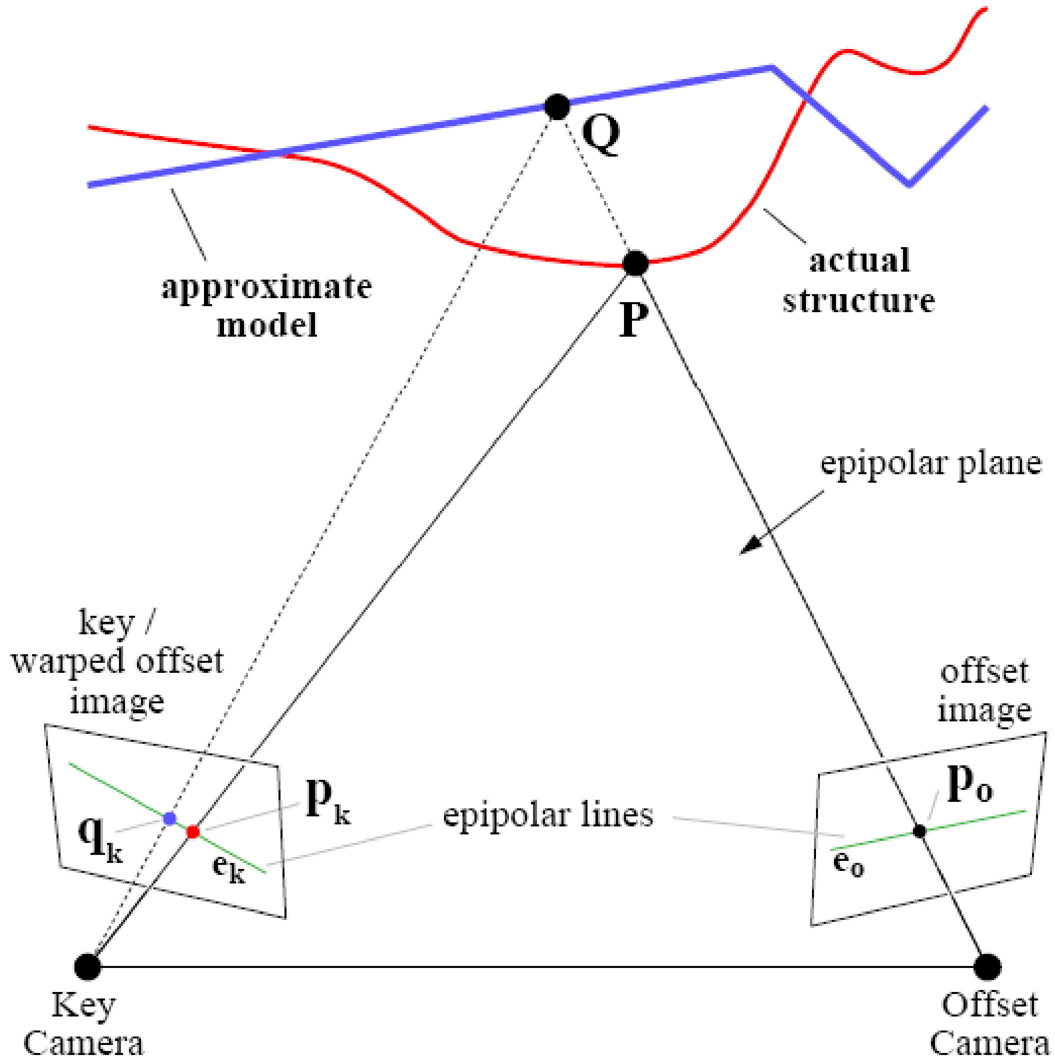


Model-based stereo

- Measuring the deviation from the approximate model.
  - Rather than measuring the structure without any prior information.
- Correlation-based matching
  - Utilizing warped offset images.



Model-based stereo (cont.)



Model-based Stereo (cont.)