3D Manipulation of 2D Images*

Peter Cho (cho@ll.mit.edu) & Alexandru Vasile (alexv@ll.mit.edu)

MIT Lincoln Laboratory

MIT IAP Course
January 2013

Course overview & class 1 notes

*This work was sponsored by the Department of the Air Force under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

Digital Imagery Explosion

- **Quantity & quality of digital imagery are rapidly increasing**
  - **Billions of photo stills & video clips are already available on web**
- **Current navigation through image repositories usually requires wading through sea of thumbnails**
  - **Little connection typically exists between thumbnails besides human-tagged keywords**
- **Current searching of photo stills & video frames is frequently frustrating**
  - **Often cannot retrieve information about features of interest**

Total number of flickr website photos vs year

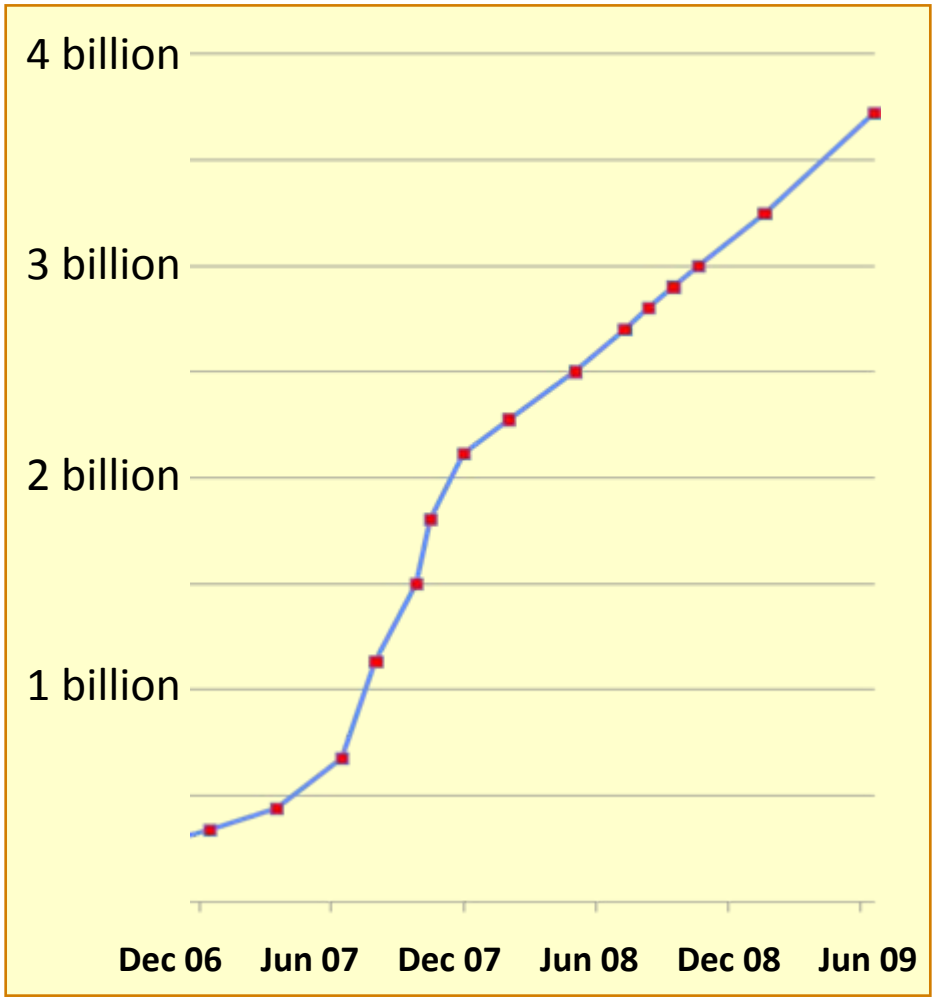
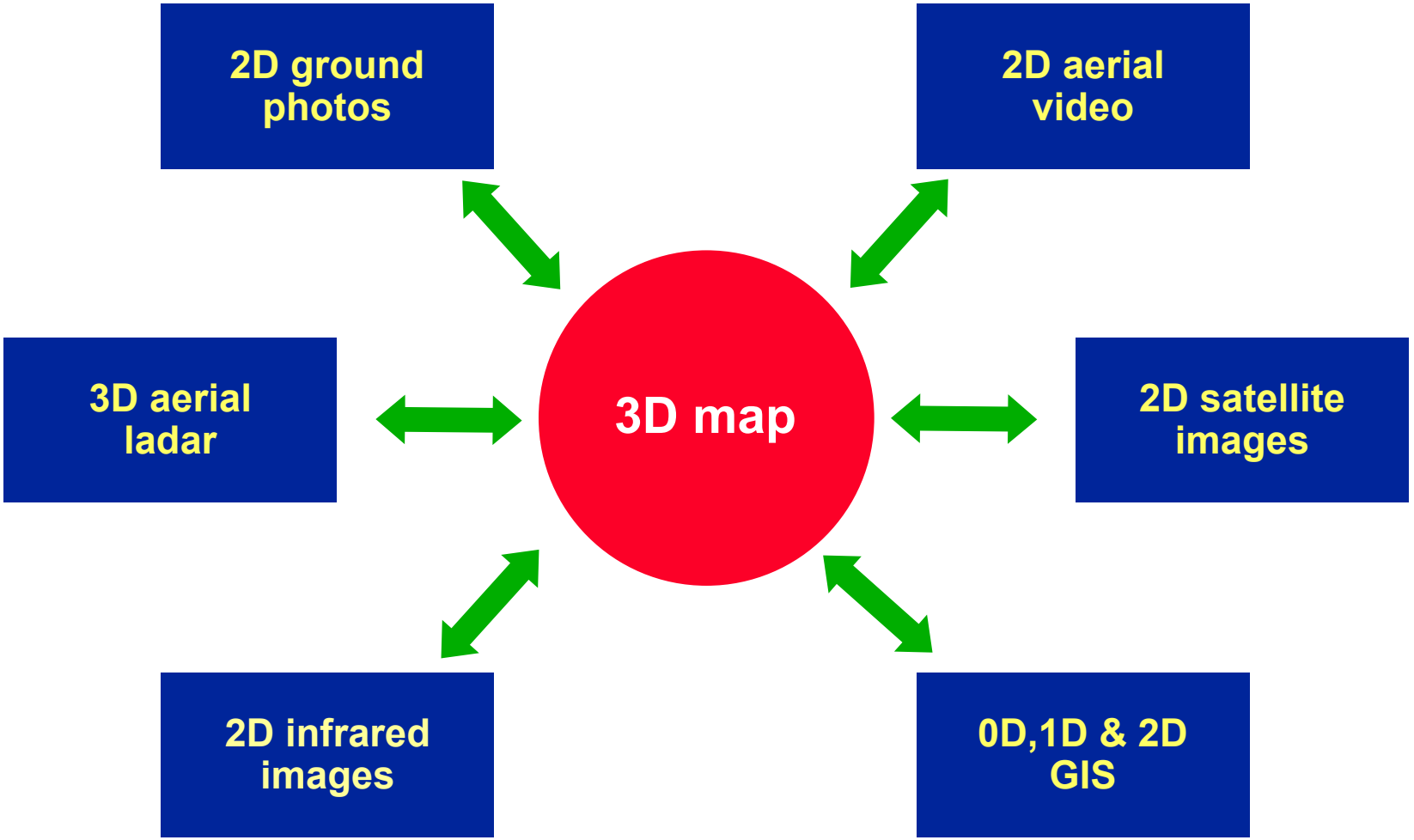


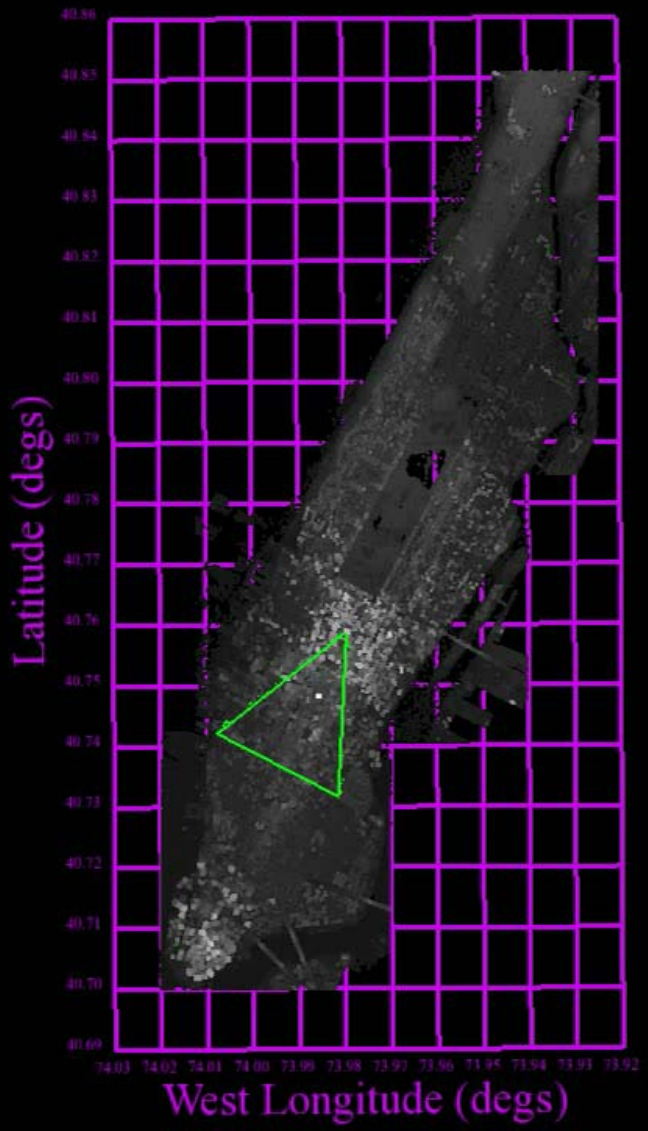
Figure from Cornell Prof. Noah Snavely

Geometrical Organization of Images



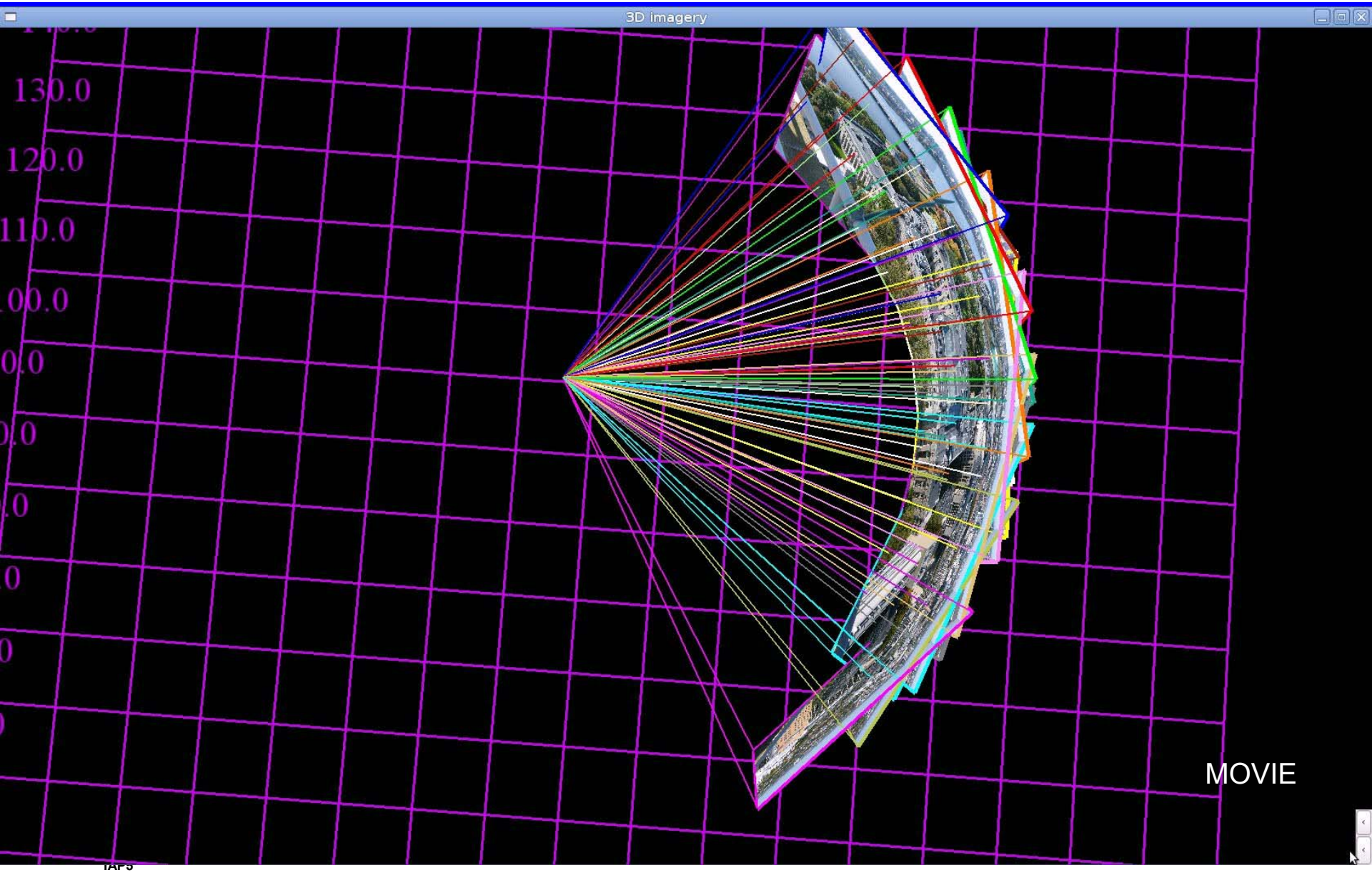
3D maps provide geometrical framework for organizing imagery collected at different times, places & perspectives

Example Application:
2D Imagery Insertion into 3D Maps (Class 1)

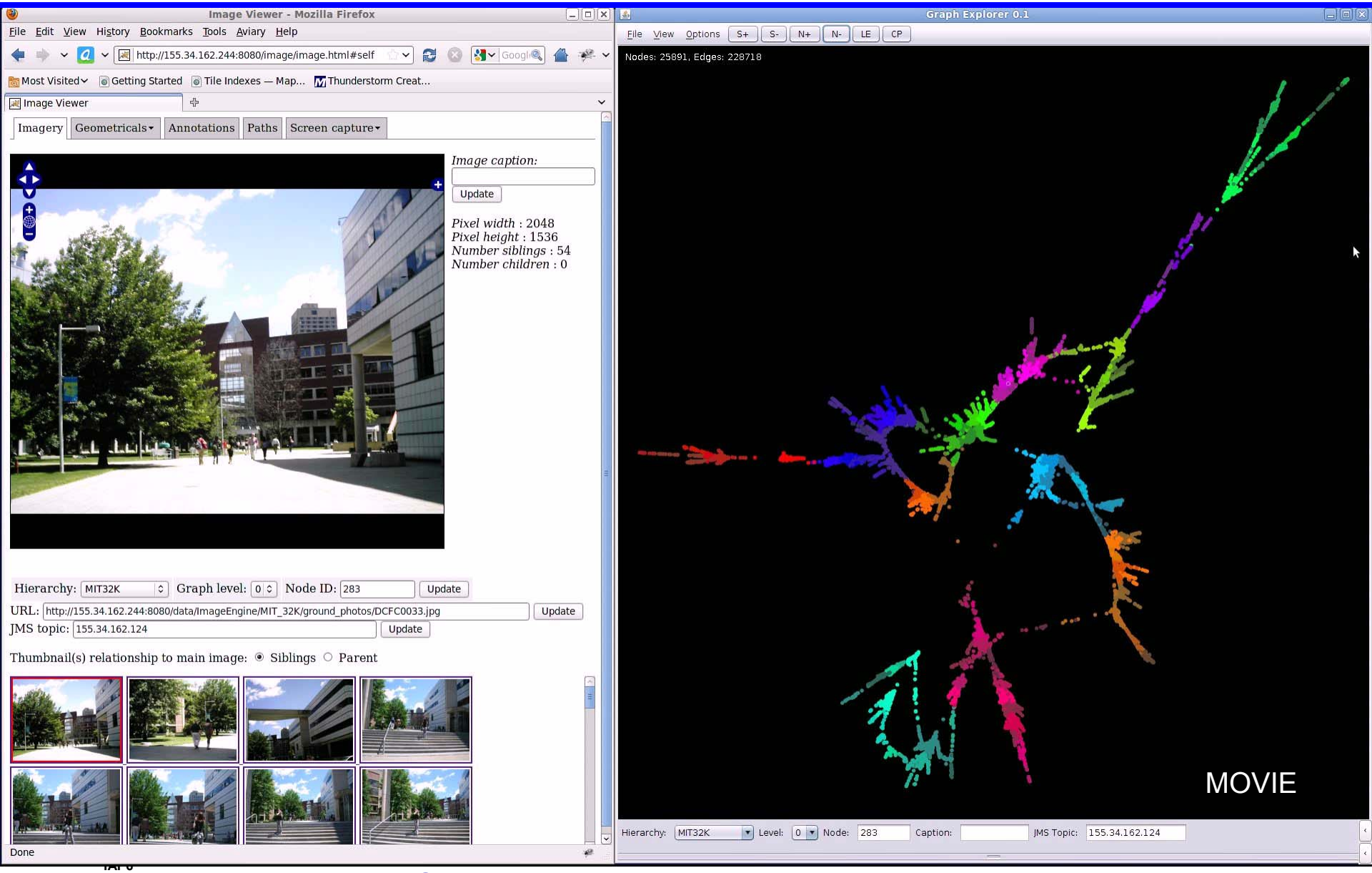


MOVIE

Example Application: Panorama Formation (Class 2)



Example Application: Automatic Image Matching (Class 3)



Graph viewer developed by Michael Yee

Example Application: 3D Reconstruction (Class 4)

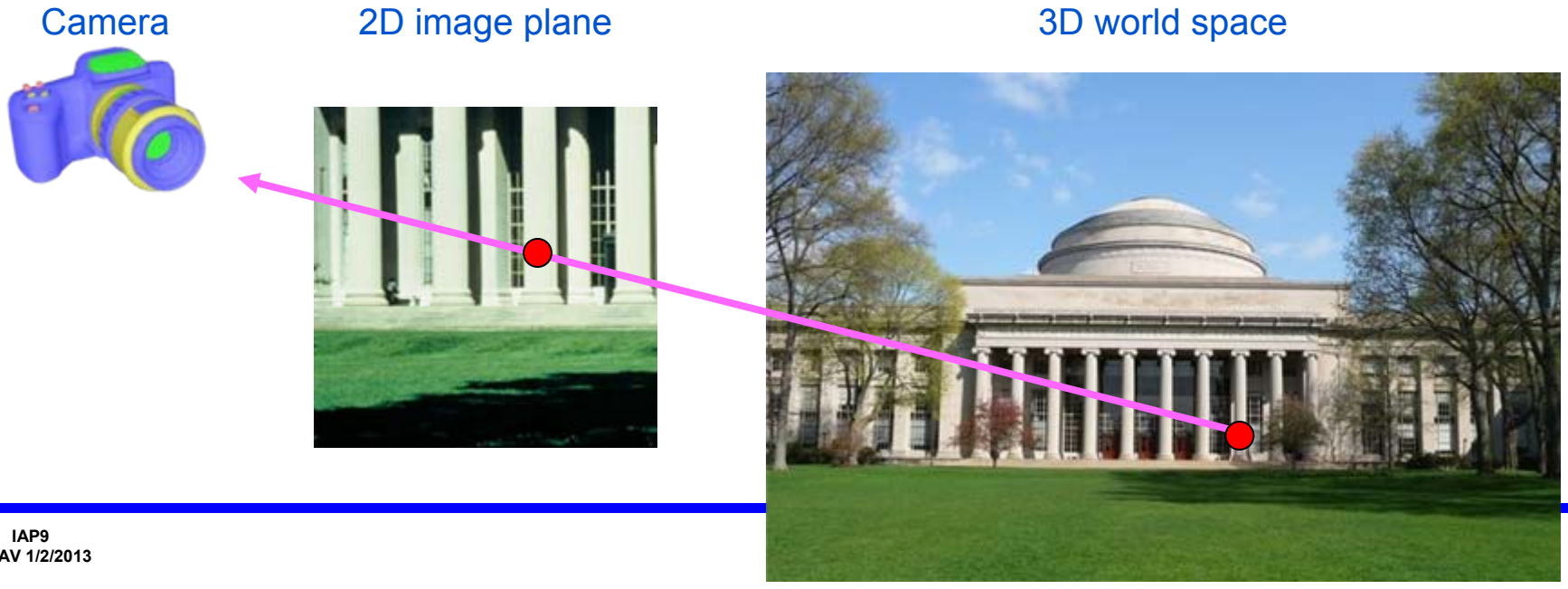
MOVIE

Course Outline

- Class 1: Single-view geometry
  - Pinhole camera model
  - 2D image insertion into 3D map
  - Camera calibration
- Class 2: Panorama formation
  - Homographies
  - 2D & 3D mosaics
  - Geometric propagation of knowledge
- Class 3: Two-view geometry
  - Fundamental matrix
  - SIFT feature matching & RANSAC
  - Epipolar geometry
- Class 4: 3D reconstruction
  - Structure from motion
  - Bundle adjustment
  - Photo tourism

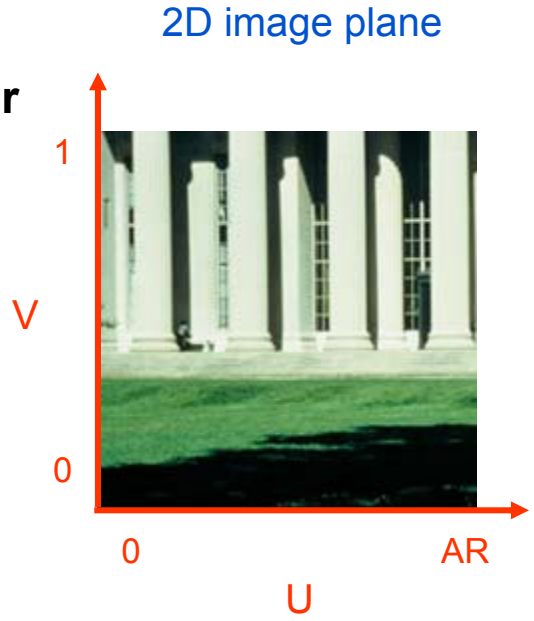
Single-View Geometry

- A camera maps the 3D world onto a 2D image plane
  - Camera images are ANGLE-ANGLE projections of world space
- World-space distances cannot be determined from pixel information alone without additional metadata
  - Hollywood frequently takes advantage of this geometrical fact
- The mathematics of camera mappings is most conveniently described in terms of projective rather than Euclidean geometry



Projective Geometry Spaces

- **Represent 2D points within camera image planes in terms of 3D homogeneous vectors**
  - **Peter's conventions:**
    - **Work with "dimensionless" coordinates u & v rather than "dimensionful" pixel coordinates px & py**
    - **u ranges from 0 to aspect ratio AR = image width/ image height**
    - **v always ranges from 0 to 1**
    - **Image plane center $(u_0,v_0)=(0.5*AR,0.5)$**
  - **(u,v,1) ~ (ku, kv, k) for $k \neq 0$ describes an equivalence class**
  - **Set of all equivalence classes in R^3-(0,0,0) forms 2D projective space P^2**



- **Represent 3D points within world-space in terms of 4D homogeneous vectors**
  - **(X,Y,Z,1) ~ (kX, kY, kZ,k) for $k \neq 0$ describes equivalence class**
  - **Set of all equivalence classes in R^4-(0,0,0) forms 3D projective space P^3**

Camera Projection Matrix

- **Projective relationship between P^3 and P^2 is LINEAR**
  - **So cameras can be simply modeled using matrix algebra if we work with projective rather than Euclidean geometry**
- **To good approximation, modern digital cameras are described in terms of 3×4 projection matrices**

Camera projection matrix

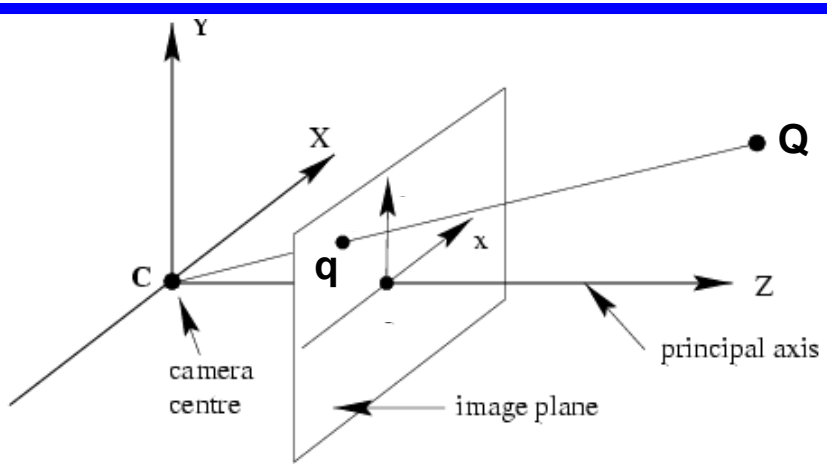
$$P_{3\times4}\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} u \\ v \\ w \end{pmatrix} \approx \begin{pmatrix} u/w \\ v/w \\ 1 \end{pmatrix}$$

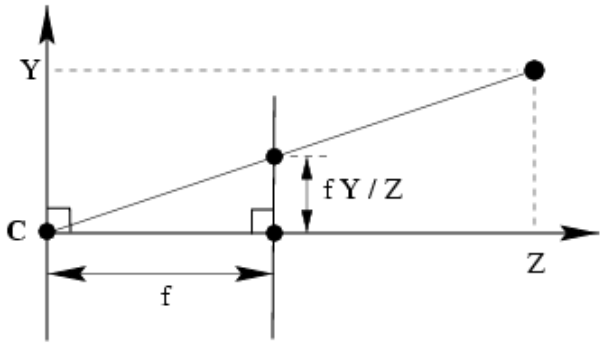
Homogeneous coordinates of corresponding point within 2D image plane

Homogenous coordinates of 3D world space point

  - **Linear relationship doesn't model small distortions in real lenses**
  - **But we won't worry about such small effects in this class…**

Pinhole Projection in Camera Frame



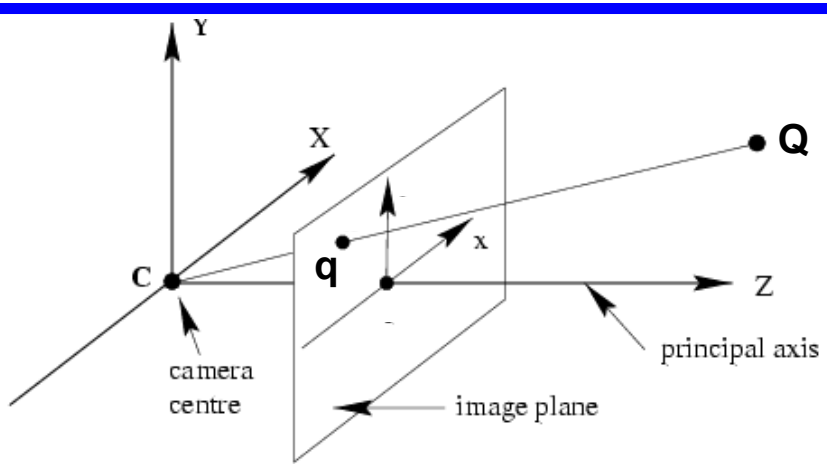


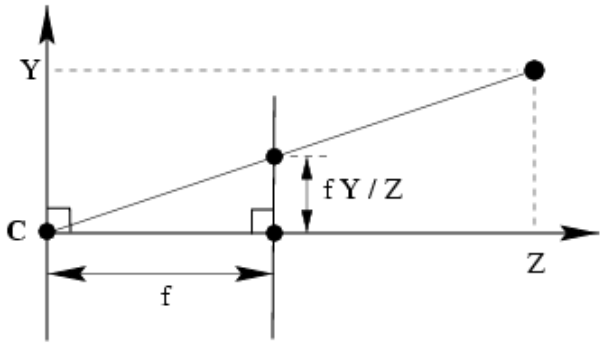
Figures from Hartley & Zisserman

- **Pinhole model maps world-space point Q to image plane point q via ray joining Q to camera center C**
  - **(X,Y,Z) projects onto (fX/Z,fY/Z,f) within image plane Z=f**
  - **In digital cameras, image plane is located behind rather than in front of C**
  - **3D → 2D projection is simple linear mapping when expressed in terms of homogeneous coordinates:**

$$\begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Pinhole Projection in Camera Frame



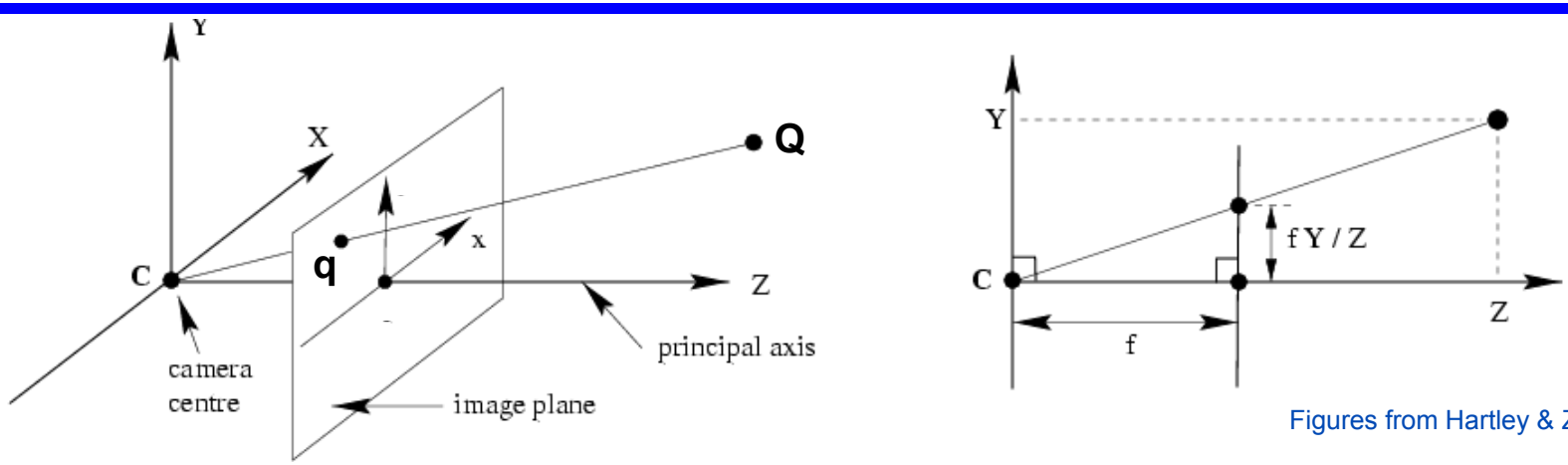


Figures from Hartley & Zisserman

- **Pinhole model maps world-space point Q to image plane point q via ray joining Q to camera center C**
  - **(X,Y,Z) projects onto (fX/Z,fY/Z,f) within image plane Z=f**
  - **In digital cameras, image plane is located behind rather than in front of C**
  - **3D → 2D projection is simple linear mapping when expressed in terms of homogeneous coordinates:**

$$\begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & & & \\ & f & & \\ & & 1 & \end{bmatrix} \begin{bmatrix} 1 & & & 0 \\ & 1 & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Pinhole Projection in Camera Frame



Figures from Hartley & Zisserman

- **Pinhole model maps world-space point Q to image plane point q via ray joining Q to camera center C**
  - **(X,Y,Z) projects onto (fX/Z,fY/Z,f) within image plane Z=f**
  - **In digital cameras, image plane is located behind rather than in front of C**
  - **3D → 2D projection is simple linear mapping when expressed in terms of homogeneous coordinates:**

Homogenous coordinates of 2D point in image plane

Homogenous coordinates of 3D point in camera frame

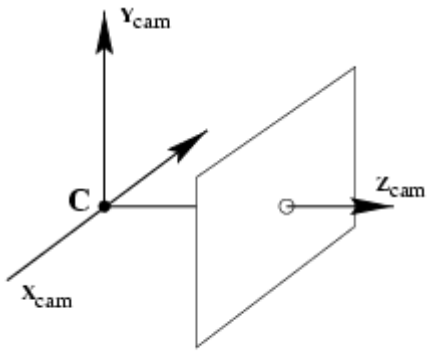
$$q = P_{3\times4} Q$$

$$P_{3\times4} = \mathrm{diag}(f, f, 1)\left[I_{3\times3} \mid 0\right]$$

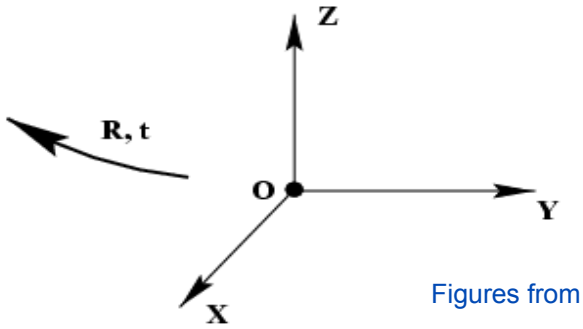
Eqn 1.1

Camera vs World Frames

Camera frame



World frame



Figures from Hartley & Zisserman

- **3D points are generally expressed in terms of world rather than camera coordinates**
  - **Two frames are related by a rotation & translation**

3×3 rotation matrix relating camera frame to world frame

$$\tilde{Q} = R\left(\tilde{Q}^{world} - \tilde{C}\right)$$

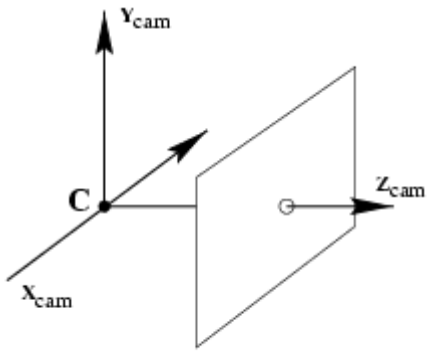
Inhomogenous coordinates of 3D point in camera frame

Inhomogenous coordinates of 3D point in world frame

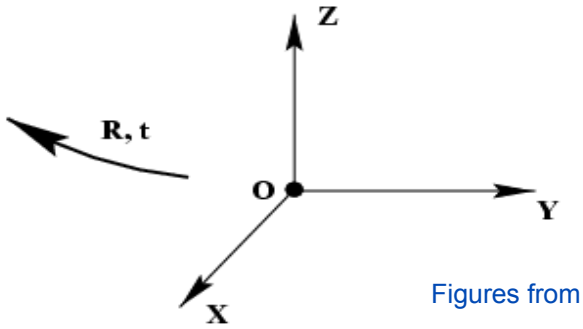
Camera center coordinates in world frame

Camera vs World Frames

Camera frame



World frame



Figures from Hartley & Zisserman

- **3D points are generally expressed in terms of world rather than camera coordinates**
  - **Two frames are related by a rotation & translation**

$$Q = \begin{bmatrix} \mathrm{R} & -\mathrm{R}\tilde{\mathrm{C}} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} \mathrm{R} & -\mathrm{R}\tilde{\mathrm{C}} \\ 0 & 1 \end{bmatrix} Q^{world}$$

Eqn 1.2

Homogenous coordinates of 3D point in camera frame

Homogenous coordinates of 3D point in world frame

Pinhole Camera Model

- **Simplest pinhole camera mapping from 3D world space to 2D image plane expressed in terms of homogeneous coordinates is obtained after combining Eqns 1.1 & 1.2:**

$$q = P_{3\times4} Q^{\text{world}}$$

where projection matrix

$$P_{3\times4} = KR\left[I_{3\times3} \mid -\tilde{C}\right] \equiv K\left[R \mid t\right] \qquad \text{Eqn 1.3}$$

- **Upper triangular "camera calibration" matrix** $K_{3\times3} = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix}$

contains 1 focal & 2 image plane center parameters

- **Rotation $R_{3\times3}$ contains 3 camera angles (azimuth, elevation & roll)**
- **Translation vector $t_{3\times1}$ contains 3 parameters**

➡ **3 "intrinsic" + 6 "extrinsic" parameters are needed for pinhole model**

2D Image Insertion into 3D Map

- **All geometry information necessary to embed a 2D image within a 3D map can be recovered from projection matrix $P_{3\times4}$**
  - **Camera position (Eqn 1.3)**
  - **Camera's pointing direction**
  - **3D orientations for camera's u and v axes**
  - **Camera's horizontal and vertical fields-of-view**

- **Once geometry information for a camera is known, 2D image may be represented as a frustum inside a 3D map**
  - **Since photos are angle-angle projections, image plane location within frustum is arbitrary**

Photo shot from Rockefeller Center

3D view "through" & "around" 2D photo

Camera Calibration from 3D/2D Tiepoint Pairs

- **Projection matrix can be determined if sufficiently many 3D/2D tiepoint pairs are known**

$$\mathrm{q_i} = \begin{pmatrix} u_i \\ v_i \\ w_i \end{pmatrix} = \mathrm{P_{3\times4}} \mathrm{Q_i^{world}} = \mathrm{P_{3\times4}} \begin{pmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{pmatrix}^{\mathrm{world}}$$

  - **Index i=1, 2, …, n labels tiepoint pairs**
  - **Drop "world" superscript from here on**

- **Since q and PQ are both homogeneous 3-vectors, they are equal only up to a nonzero constant**
  - **But their cross-product must vanish**

$$\mathrm{q_i} \times \mathrm{P_{3\times4}} \mathrm{Q_i} = 0$$

Eqn 1.4

Camera Calibration from 3D/2D Tiepoint Pairs

- **Manipulate Eqn 1.4 using tensor notation:**

Fully antisymmetric epsilon symbol

$$0_a = \sum_{b,c=1}^{3} \varepsilon_{abc} q_i^b (PQ_i)^c = \sum_{b,c=1}^{3} \varepsilon_{abc} q_i^b \underbrace{p^{(c)T}}_{c^{th}\text{ row of } P_{3x4}} Q_i$$

$$= \sum_{b,c=1}^{3} \varepsilon_{abc} q_i^b Q_i^T p^{(c)}$$

- **Explicitly rewrite last relation in terms of homogeneous 3D & 2D point coordinates:**

$$\underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & -w_i X_i & -w_i Y_i & -w_i Z_i & -w_i & v_i X_i & v_i Y_i & v_i Z_i & v_i \\ w_i X_i & w_i Y_i & w_i Z_i & w_i & 0 & 0 & 0 & 0 & -u_i X_i & -u_i Y_i & -u_i Z_i & -u_i \end{bmatrix}}_{\text{2×12 matrix containing i}^{th}\text{ 3D/2D tiepoint pair}} \underbrace{\begin{bmatrix} p^{(1)} \\ p^{(2)} \\ p^{(3)} \end{bmatrix}}_{\text{12×1 matrix containing entries of } P_{3x4}} = 0$$

2×1

Eqn 1.5

Camera Calibration from 3D/2D Tiepoint Pairs

- **For n sets of 3D/2D tiepoint pairs**

$$(X_i, Y_i, Z_i) \leftrightarrow (u_i, v_i, w_i = 1) \quad \text{with i=1,2,…n}$$

stack up LHS of Eqn 1.5 into a 2n × 12 data matrix:

$$A_{2n\times 12}\ P_{12\times 1} = 0_{2n\times 1} \qquad \text{Eqn 1.6}$$

Data matrix containing known tiepoint pairs

Unknown 3×4 camera projection matrix expressed as a column vector

- **Data matrix A inevitably contains errors. So employ Singular Value Decomposition to find least-squares solution to Eqn 1.6**
  - **Rule of thumb: 10-20 tiepoint pairs randomly spread across image are needed to calibrate an *a priori* unknown camera with reasonable accuracy**

Camera Projection Matrix from SVD

- Singular Value Decomposition of data matrix A yields camera projection matrix

$$A_{2n\times 12} = U_{2n\times 12}\, D_{12\times 12}\, V_{12\times 12}^{T} \qquad \text{Eqn 1.7}$$

U is a matrix with orthogonal column vectors, D is a diagonal matrix with non-negative entries & V is an orthogonal matrix

- Least squares solution to Eqn 1.6 is given by column of V corresponding to smallest singular value in D
  - Arrange SVD so that diagonal entries of D appear in descending order
  - Then $P_{12\times 1}$ equals last column in V

- Rearrange column vector $P_{12\times 1}$ into camera projection matrix $P_{3\times 4}$

Computer Lab 1:
Deduce Mystery Camera Geolocation

Q: What are the latitude, longitude & altitude geocoordinates for the cameras which shot these two photos?

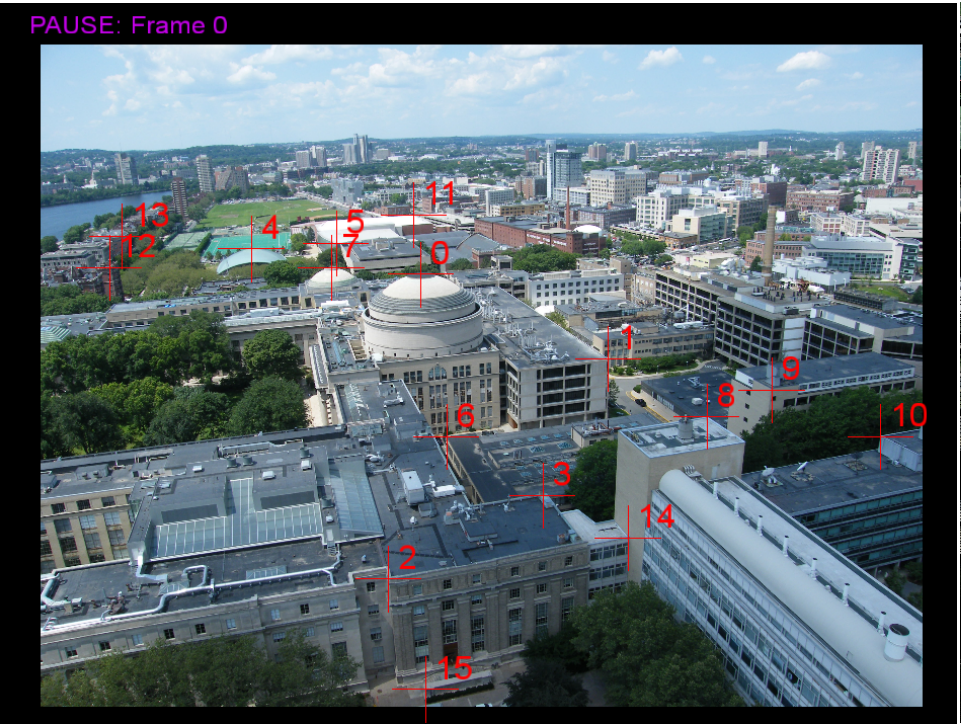
Photo 1: “MIT”

Photo 2: “Copley”

“MIT” Tiepoint Inputs

- **Given manually selected set of tiepoint pairs for photos**
  - **3D (X,Y,Z) points are specified in Universal Transverse Mercator (UTM) geocoordinates**
  - **2D (u,v) counterparts are in Peter’s image plane coordinate system**

2D features selected from “MIT” photo

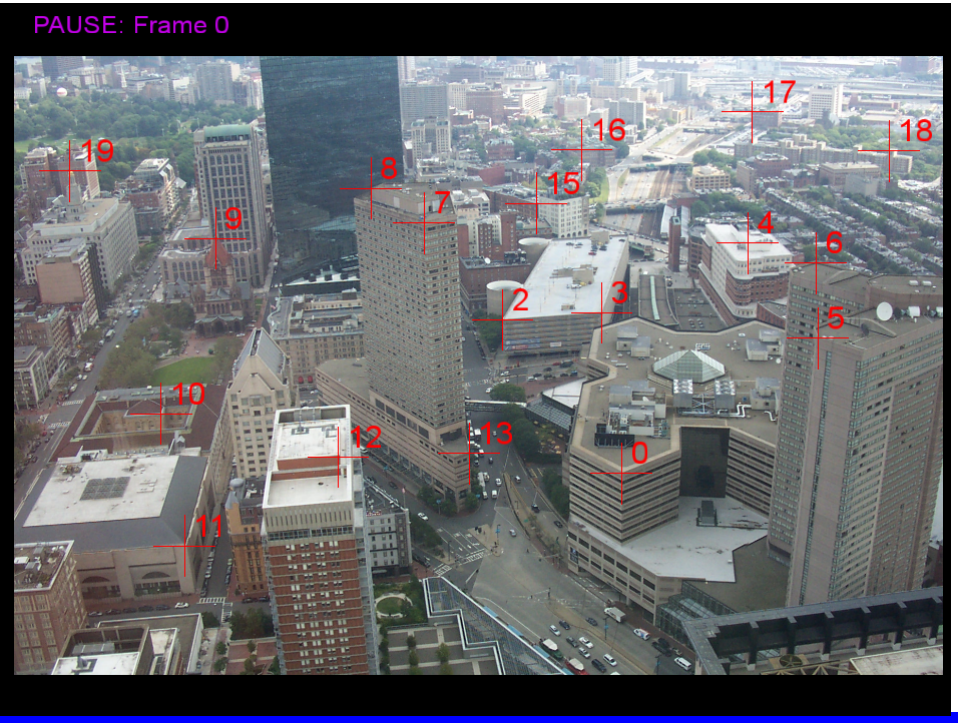


Corresponding 3D features in laser radar map

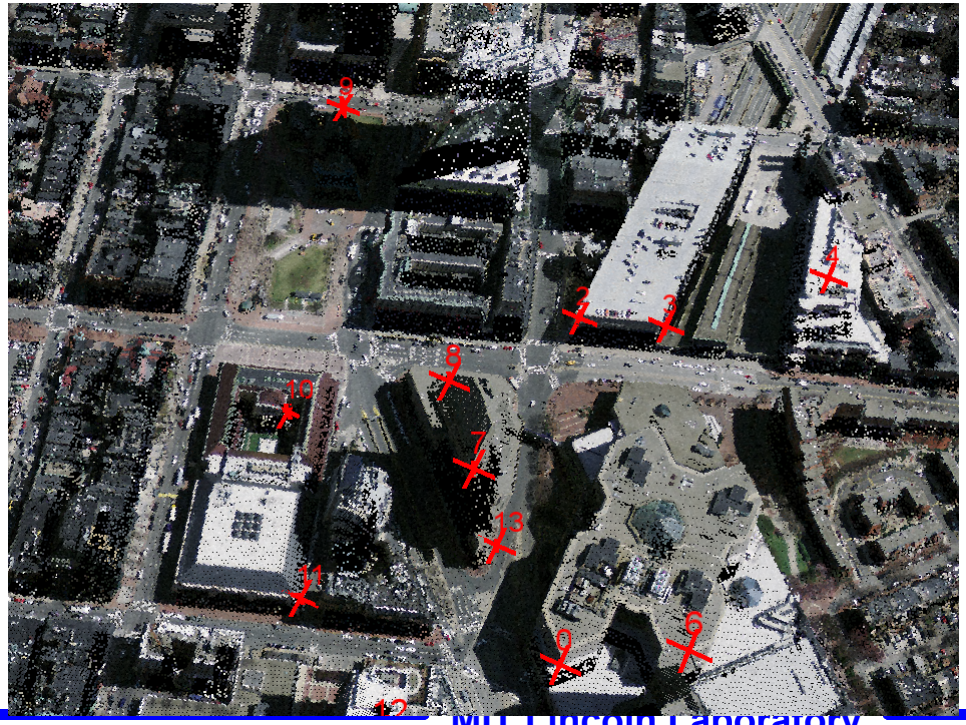
“Copley” Tiepoint Inputs

- **Given manually selected set of tiepoint pairs for photos**
  - **3D (X,Y,Z) points are specified in Universal Transverse Mercator (UTM) geocoordinates**
  - **2D (u,v) counterparts are in Peter’s image plane coordinate system**

2D features selected from “Copley” photo



Corresponding 3D features in laser radar map



Computer Lab Instructions

- Download text files containing 3D/2D tiepoints for the "MIT" & "Copley" photos
  - http://web.mit.edu/alexv/Public/IAP_2013/Class1/MITtiepoints.txt
  - http://web/mit.edu/alexv/Public/IAP_2013/Class1/Copleytiepoints.txt
- Set up data matrix A for either "MIT" or "Copley" data set (Eqns 1.5 & 1.6)
- Compute the singular value decomposition of A (Eqn 1.7)
- Recover least-squares solution for camera projection matrix (Eqn 1.6)
- Extract camera's geolocation from its projection matrix (Eqn 1.3)
  - No need to solve for K & R individually in order to find $\tilde{C}$
  - Convert from UTM to lat-lon-altitude using on-line web utilities
  - Look up deduced camera geolocation on Google Earth

Annotated References for Class 1

- D.A. Forsythe & J. Ponce, "Computer vision: A modern approach", Prentice Hall, 2003.
  - This textbook provides a gentle introduction to the pinhole model & camera calibration in chapters 2 and 3.

- R. Hartley and A. Zisserman, "Multiple view geometry in computer vision (2nd edition)", Cambridge University Press, 2003.
  - This standard text for computer vision researchers goes rather heavy on math. Chapters 6 & 7 cover camera models and calibration.

- J. Zhu, "3D from Pictures", www.cs.virginia.edu/~jz8p/.../Structure from motion.ppt, 2006.
  - There are many multi-view geometry lectures on the web. Here is just one example which covers some of the material for this class.