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

Class 2 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.

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

Geometry Transformation Problems

- **3D to 2D camera projection (class 1)**
  - Given a set of points (X,Y,Z) in world-space and a corresponding set of points (U,V) in a camera's image plane, find the projection that maps 3D to 2D
- **2D homography (class 2)**
  - Given a set of points (U,V,1) in 2D projective space P^2 and a corresponding set of tiepoints (U',V',1) also in P^2, compute the projective transformation that maps the former into the latter
- **Fundamental matrix (class 3)**
  - Given a set of points (U,V) in one image and a matching set of tiepoints (U',V') in another image, compute the "fundamental" matrix consistent with these correspondences
  - Fundamental matrix relationships hold more generally than do homography mappings
- **Least-squares approach to solving for these geometry transformations is similar for all 3 cases**
  - Nonlinear methods are much more complicated to implement but yield refined transformation solutions

Homographies

- **A "homography" is a linear transformation on homogeneous 3-vectors represented by a non-singular 3×3 matrix H**

$$\begin{pmatrix} U' \\ V' \\ W' \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} U \\ V \\ W \end{pmatrix} \quad \text{or} \quad q' = Hq \qquad \text{Eqn 2.1}$$

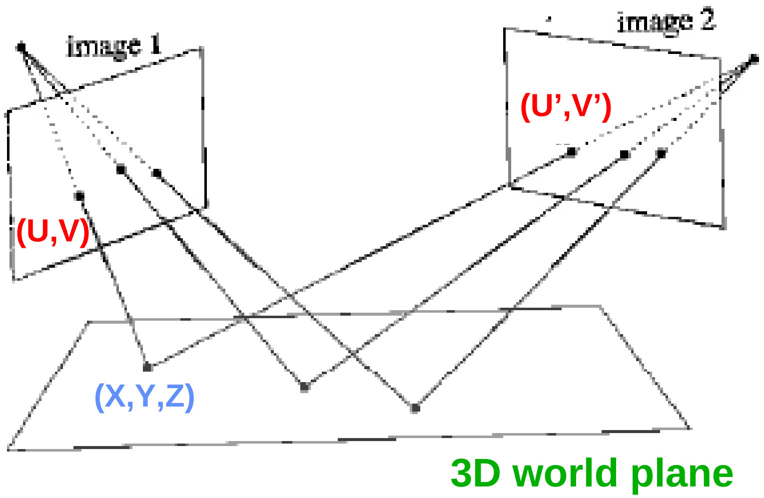
  – **Homographies are also called projective transformations, projectivities and collineations**

- **H may be multiplied by an arbitrary non-zero scale factor without altering the projective transformation**
  – **Matrix H is called "homogeneous", for only the ratios of its elements are significant (analogous to homogeneous vectors)**
  – **Homographies consequently have 8 degrees of freedom**

Homography Examples

- **Consider two images of points lying within some 3D world-plane**
  - **Mapping between their 2D homogeneous image plane coordinates is a homography**

Figure from Hartley & Zisserman

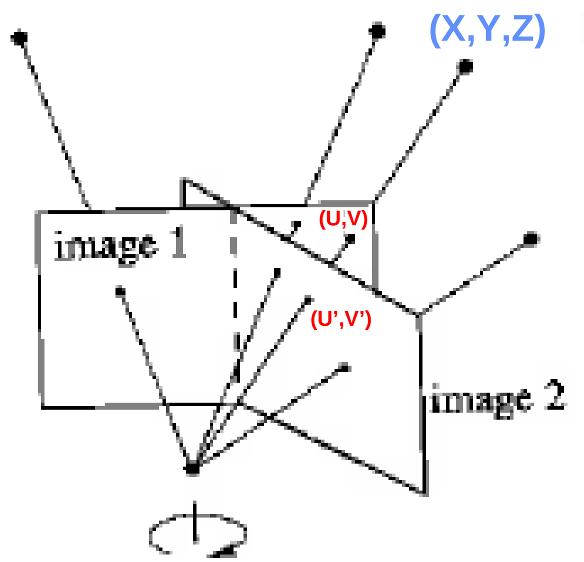


Homography Examples

- Consider two images of points lying within some 3D world-plane
  - Mapping between their 2D homogeneous image plane coordinates is a homography

- Consider two photos shot by some camera which can rotate but not translate (e.g. it is fixed atop a tripod)
  - Corresponding points within the 2 images are related by a homography

Figure from Hartley & Zisserman

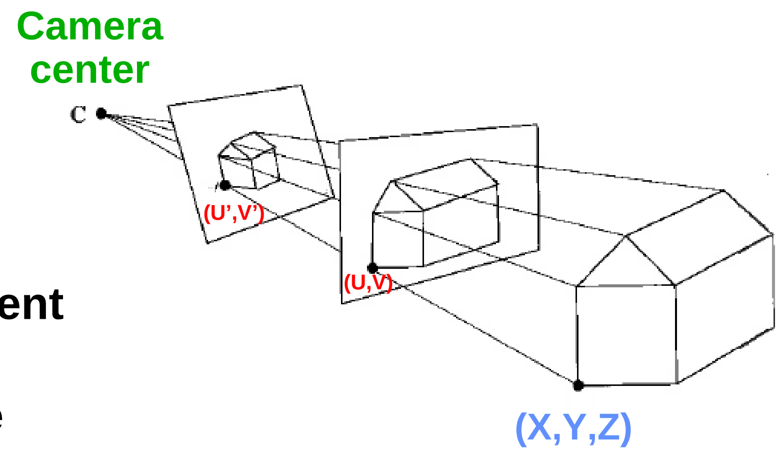


Rotating camera

Homography Examples

- Consider two images of points lying within some 3D world-plane
  - Mapping between their 2D homogeneous image plane coordinates is a homography
- Consider two photos shot by some camera which can rotate but not translate (e.g. it's fixed atop a tripod).
  - Corresponding points within the 2 images are related by a homography
- **Consider a cone of rays whose vertex corresponds to some camera's center**
  - **Different images of the world scene are generated by intersecting cone with different planes**
  - **Corresponding points in image planes are related by a homography**

Figure from Hartley & Zisserman



Homography Induced by Camera Rotation

- **Consider a 3D world-point seen by a camera on a tripod**
  - **Without loss, take camera's center to lie at world coordinate origin & align camera's axes with world directions**
  - **Extrinsic part of camera's 3×4 projection matrix assumes trivial form**

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = K\left[I_{3\times3} \mid 0\right]\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

- **Reorient tripod camera through some rotation**
  - **If 3D world-point lies within rotated camera's field-of-view, it now projects to new image plane point**

$$\begin{pmatrix} U' \\ V' \\ W' \end{pmatrix} = K\left[R \mid 0\right]\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

- **Consequently the two image plane points are related as (U',V',W')=H (U,V,W) with**

$$H = KRK^{-1}$$

Homography Computation from 2D Tiepoint Pairs

- **Homography relationship (Eqn 2.1) holds only up to an overall scale factor since q, q' & H are homogeneous**
  - **But** $\mathrm{q}'\times \mathrm{H}\mathrm{q} = 0$ **always vanishes**

- **Perform very similar manipulations on this cross product identity as those which were applied to Eqn 1.4** $\mathrm{q_i} \times \mathrm{P_{3\times 4}} \mathrm{Q_i} = 0$ **in class 1 notes**

- **Obtain homography analog to Eqn 1.5 from class 1 notes:**

$$\underbrace{\begin{bmatrix} 0 & 0 & 0 & -w_i'U_i & -w_i'V_i & -w_i'W_i & v_i'U_i & v_i'V_i & v_i'W_i \\ w_i'U_i & w_i'V_i & w_i'W_i & 0 & 0 & 0 & -u_i'U_i & -u_i'V_i & -u_i'W_i \end{bmatrix}}_{\text{2×9 matrix containing } i^{th} \text{ 2D/2D tiepoint pair}} \underbrace{\overset{3\times 1}{\begin{pmatrix} h^{(1)} \\ h^{(2)} \\ h^{(3)} \end{pmatrix}}}_{\text{9×1 matrix containing entries of } H_{3x3}} = 0$$

Eqn 2.2

Homography Computation from 2D Tiepoint Pairs

- **For n ≥ 4 sets of 2D/2D tiepoint pairs**

$$(U_i, V_i, W_i = 1) \leftrightarrow (u'_i, v'_i, w'_i = 1)$$ **with i=1,2,…n**

stack up LHS of Eqn 2.2 into a 2n × 9 data matrix:

$$A_{2n\times 9}\, h_{9\times 1} = 0_{2n\times 1}$$ **Eqn 2.3**

Data matrix containing known tiepoint pairs

Unknown 3×3 homography matrix expressed as a column vector

- **Construct Singular Value Decomposition $A = U\,D\,V^T$ where diagonal matrix D has its non-negative entries arranged in descending order**
  - **h in Eqn 2.3 then equals last column of V**
- **In practice, reasonable homographies can be found from just 4 tiepoint pairs**
  - **But estimation errors generally diminish with increasing n**

2D Mosaic Generation

- **Recall photos acquired by rotating camera are related by homographies**

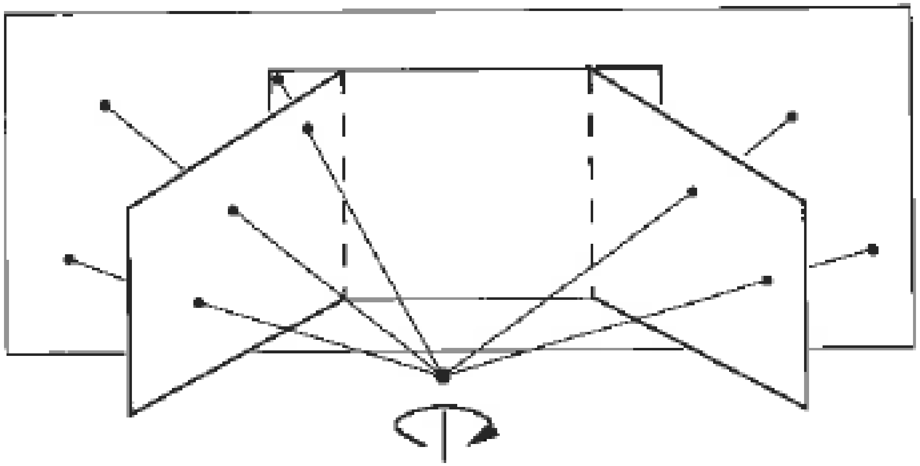
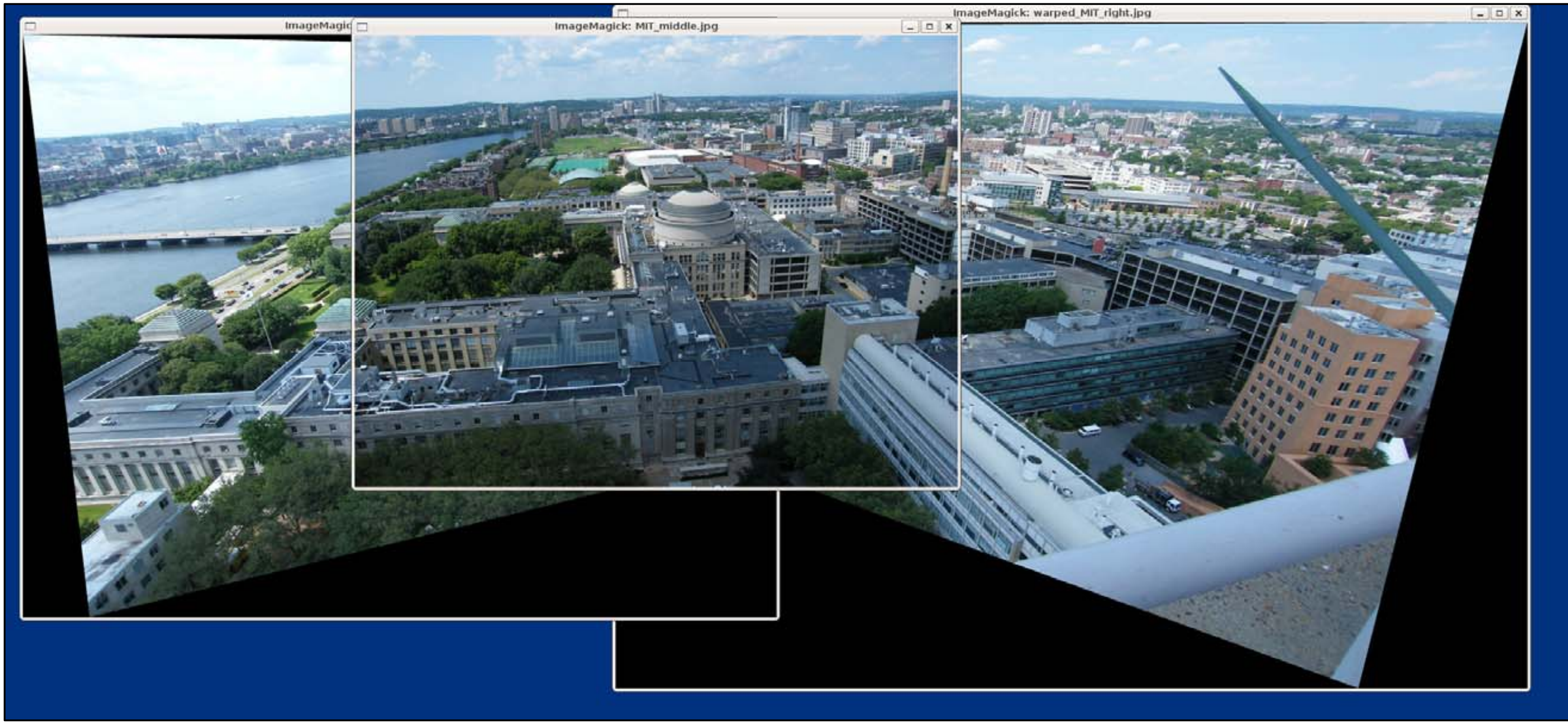


Figure from Hartley & Zisserman

- **A set of N such images may be warped to form a 2D panorama**
  - **Compute homography H_{1C} which maps first image to center image**
  - **Use H_{1C} to transform (U',V') points in first image to (U,V) coordinate system for center image**
  - **Augment center image's (U,V) content with non-overlapping part of transformed first image**
  - **Repeat steps for remaining N-2 photos**
- **The 2D panorama resulting from this procedure has a "bow-tie" shape**

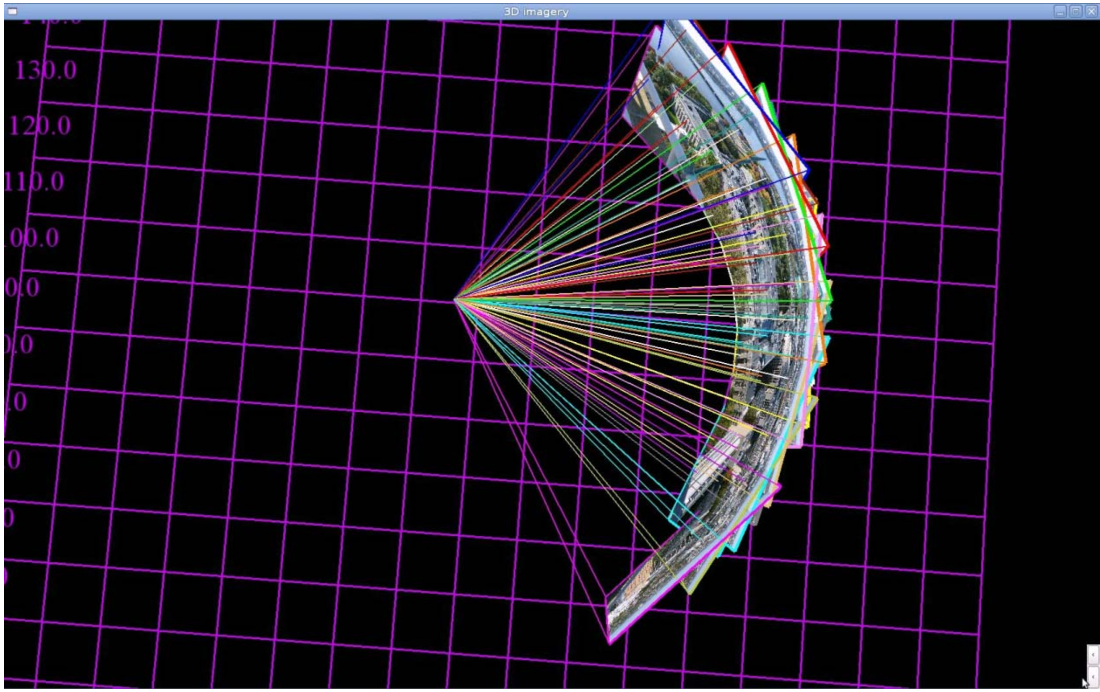
2D Mosaic Example



3D Panorama Generation

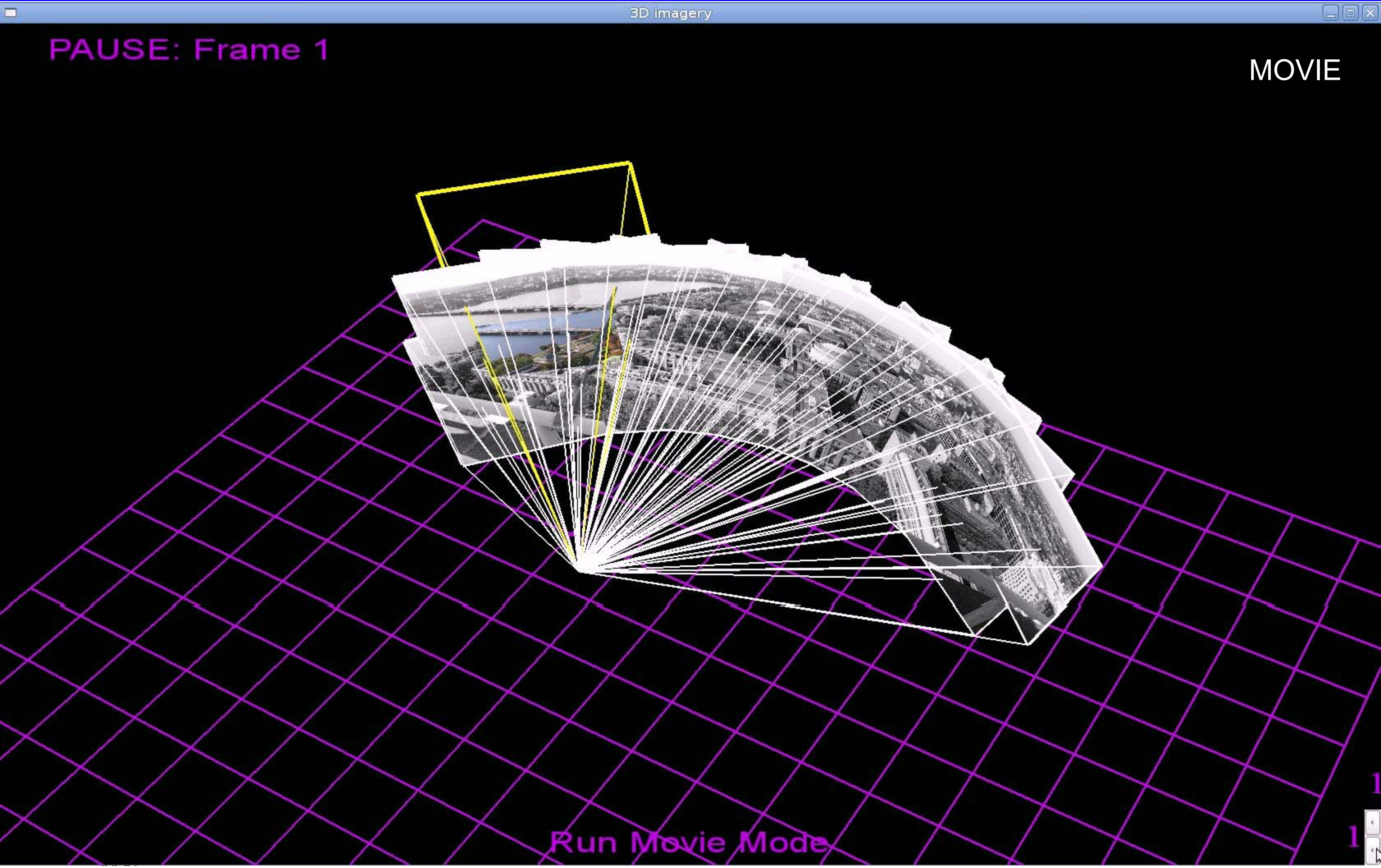
- **Generating 3D panoramas in angle-angle space rather than 2D mosaics in flat planes avoids "bow-tie" distortions**
- **Least-squares techniques provide initial estimates for camera focal & rotation matrix parameters**
- **Refine linear camera parameter estimates via non-linear optimization called "bundle adjustment"**

Reconstructed view frusta for 21 photos shot from MIT's Green Building



MOVIE

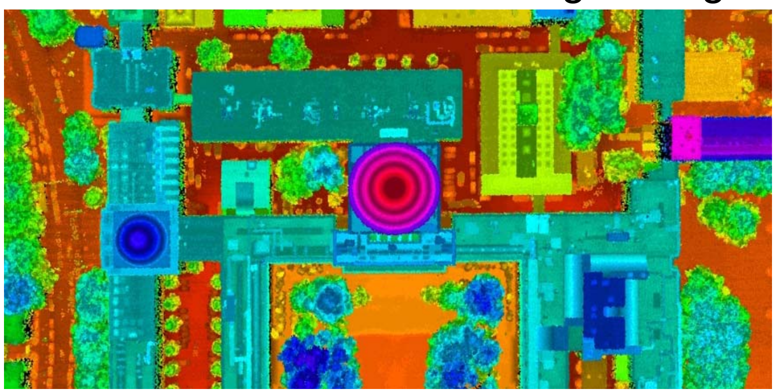
Dynamic Video Alignment with Static Mosaic



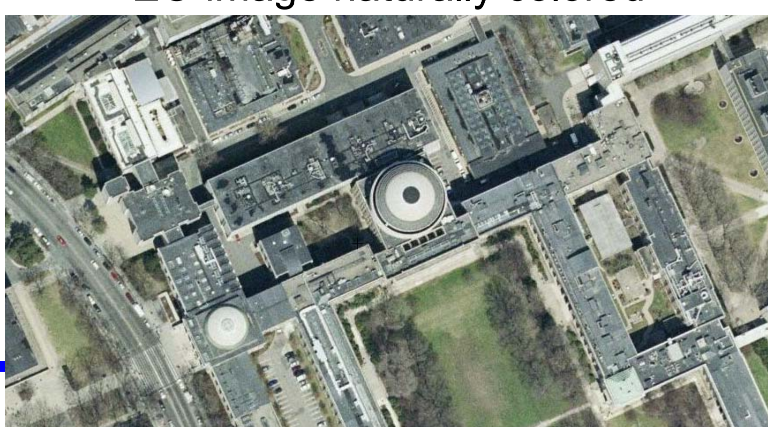
Aerial Ladar Maps

- **Computer vision cannot determine mosaics' position, orientation or scale**
  - **Need 3D world map to fix global parameters**
- **Aerial ladars efficiently measure 3D urban structure at sub-meter resolution**

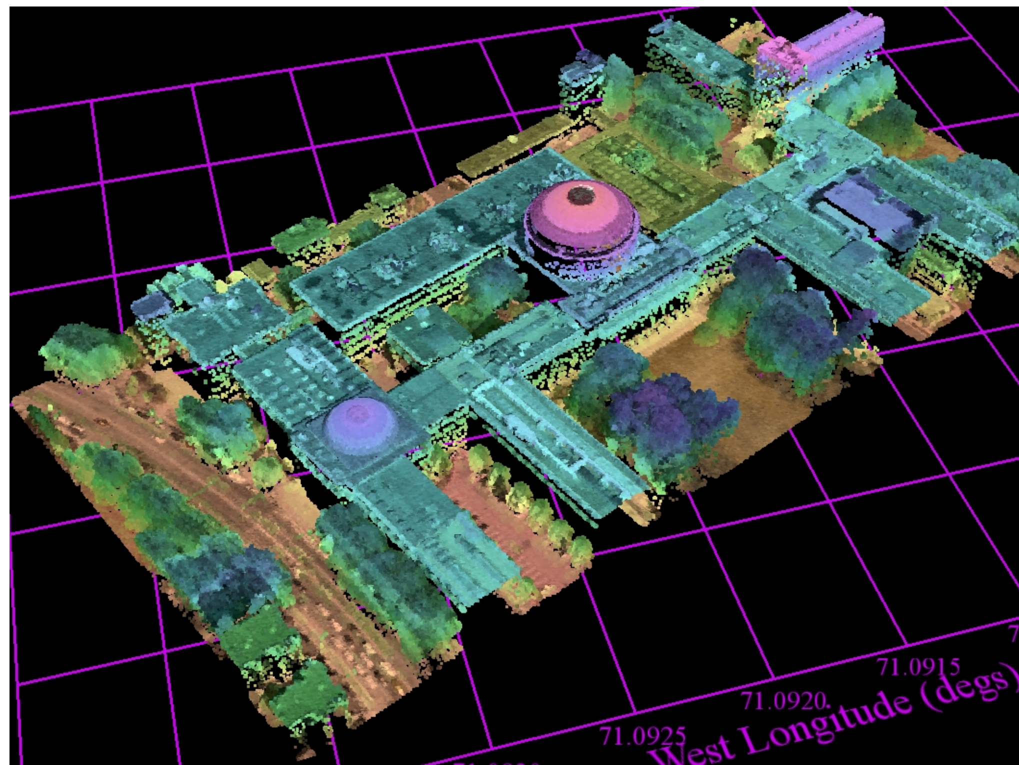
Ladar cloud colored according to height



EO image naturally colored



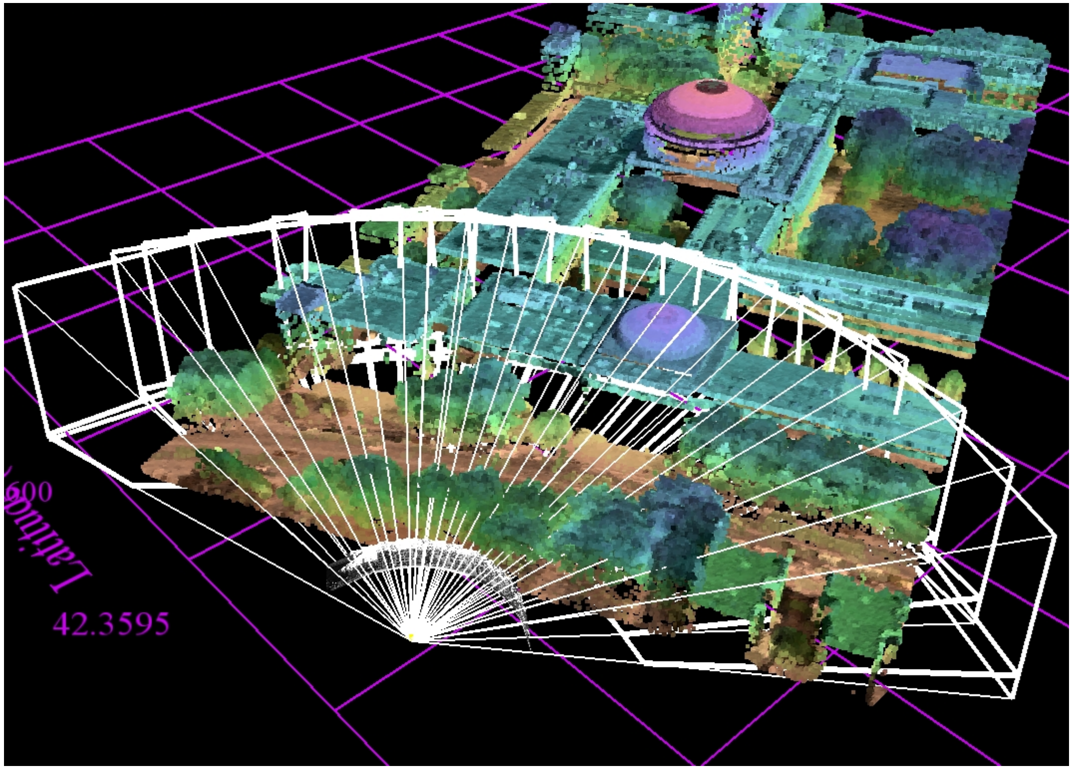
Fused 3D map on absolute world grid



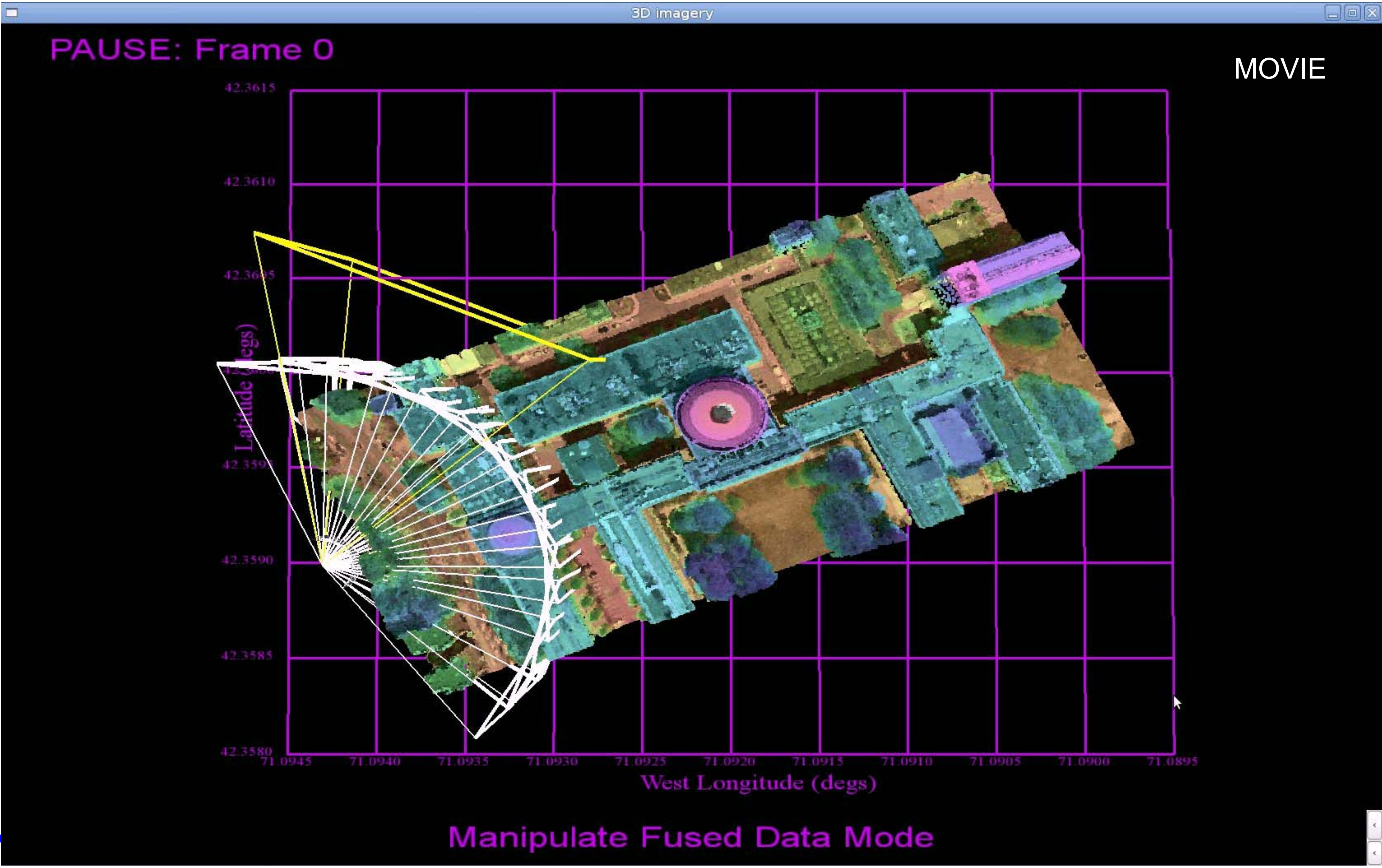
Panorama Alignment with Ladar Map

- **Quasi-randomly pick ladar points & counterparts in calibrated photos**
- **Compare opening angles between backprojected panorama & world-space rays**
- **Rescale panorama by opening angles' mean ratio**
  - **Must also adjust relative rotation between each pair of mosaiced photos**
- **Compute panorama's global rotation from SVD of corresponding world-space & image-space ray outerproducts sum**

Ground-level panorama georegistered with 3D map



Match between 3D Aerial & 2D Ground Imagery



Urban Knowledge Propagation

- **Knowledge attached to 3D voxels automatically transfers into 2D image planes once panorama is aligned with ladar point cloud**
  - **e.g. GIS layers provide longitudes & latitudes for names of buildings & streets**

- **Project geotagged information into calibrated photos via their camera model matrices**

Building & street name geotags

Annotations projected into georegistered mosaic

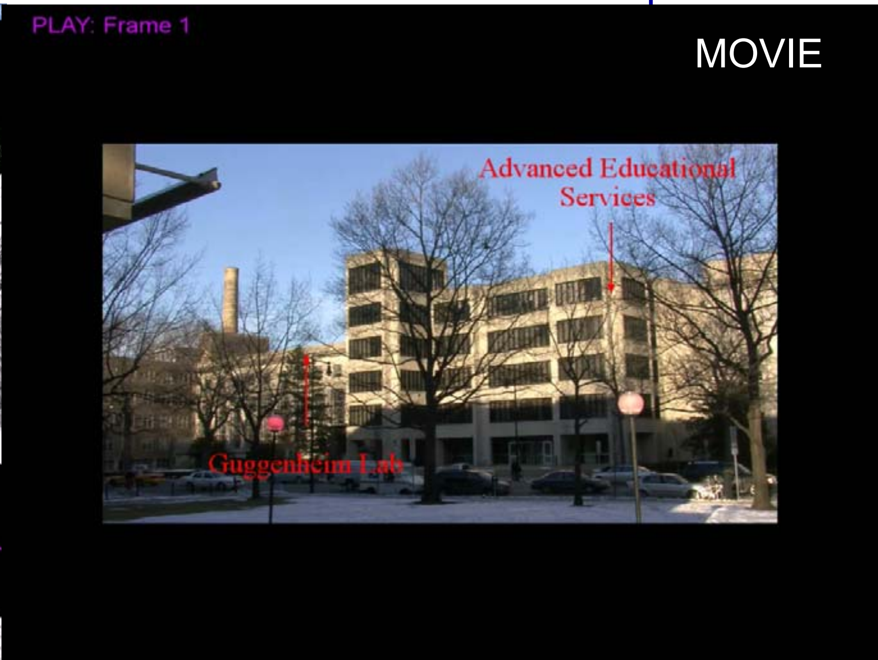
Dynamic Video Labeling

- **Annotating urban video is possible for clips georegistered with 3D map**
  - **Follow same ray matching procedure for 2nd ground-level video as for 1st skyscraper example**
  - **Ray matching insensitive to foreground traffic in video**

- **Building & street names project directly from world-space into moving video stream**
  - **Annotations track urban structures' moving image plane locations up to residual low-frequency jitter**

Video clip aligned with georegistered mosaic

Annotated video clip

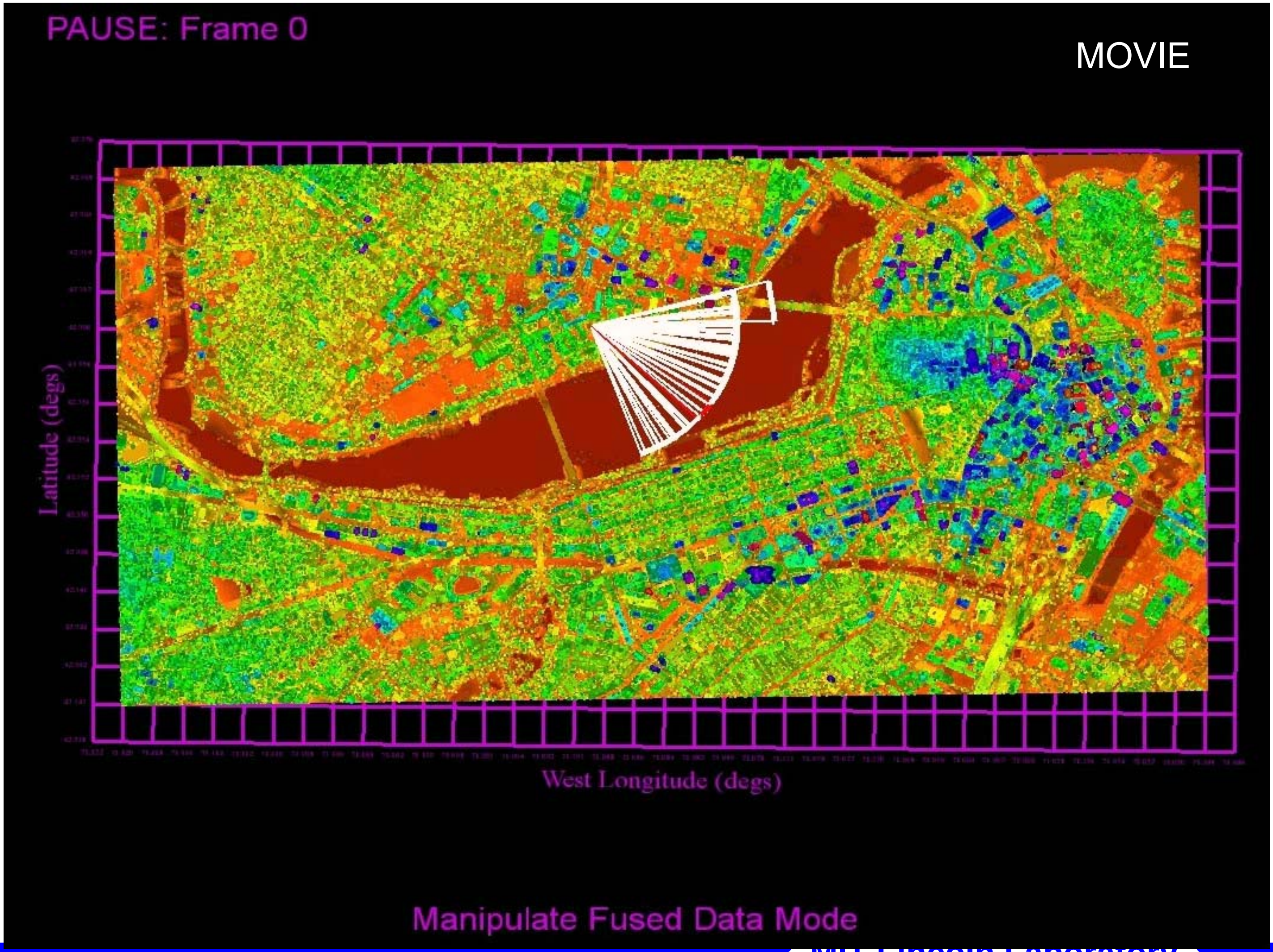


2D Digital Imagery Insertion into 3D Boston Ladar Map

- **Construct 3D mosaic of 73 Boston skyline photos & video clip shot in 2009 from MIT's Green Building**

- **Georegister panorama & video sequence to 2004 aerial ladar map**

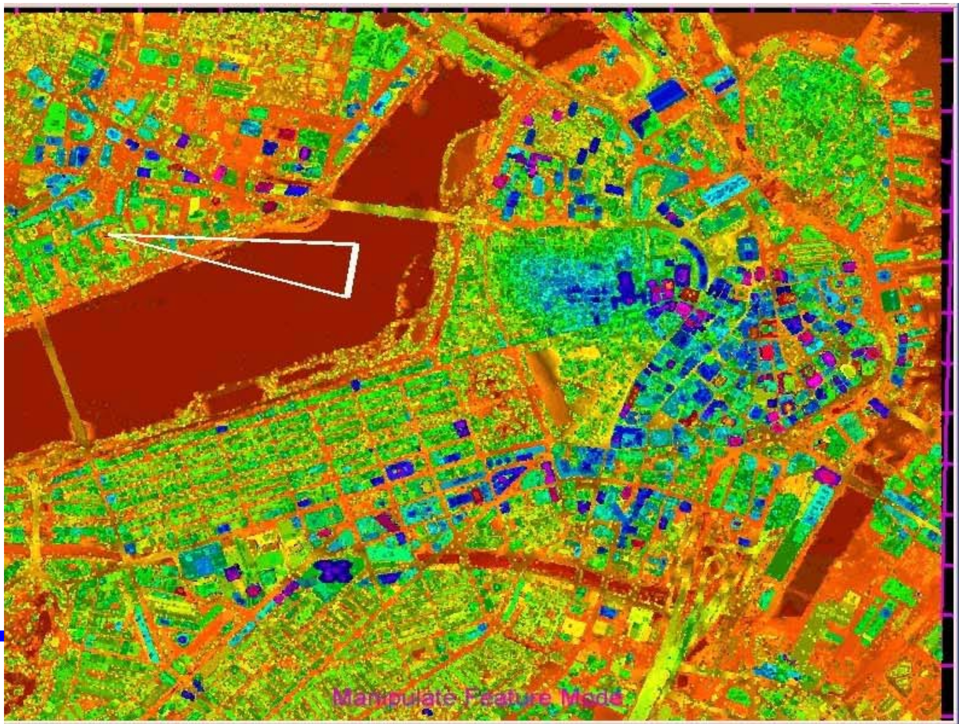
Boston skyline mosaic & video clip georegistered with aerial ladar map



Querying Urban Video Features

- **Compute 3D voxel corresponding to user-selected 2D video pixel**
  - **Ray traced ladar point determined from angle to camera's LOS & range to camera's geolocation**
- **Reproject 3D voxel back into all 2D video frames in which it is visible**
  - **Reprojection guarantees valid relationship between feature's 3D & 2D coordinates**
- **Return urban features' ranges & altitudes above sea-level**

3D ladar window



2D video window

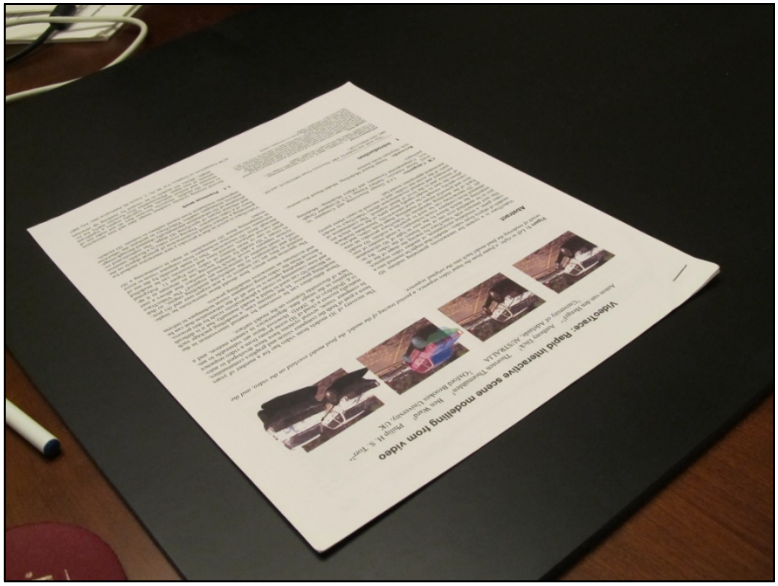
Computer Lab 2A: Orthorectification

- **Download photo of 8.5” × 11” paper sitting on a tabletop from http://web.mit.edu/alexv/Public/IAP_2013/Class2/UpsideDownPaper.jpg**
  - **Paper’s highly slanted orientation relative to camera makes its text difficult to read**
- **Download features text file which contains paper’s corners in Peter’s 2D image plane coordinates. It also contains simple, new coordinates for the paper’s corners in a rectified frame in which reading the paper’s text should be easy:**

  http://web.mit.edu/alexv/Public/IAP_2013/Class2/UpsideDownFeatures.txt

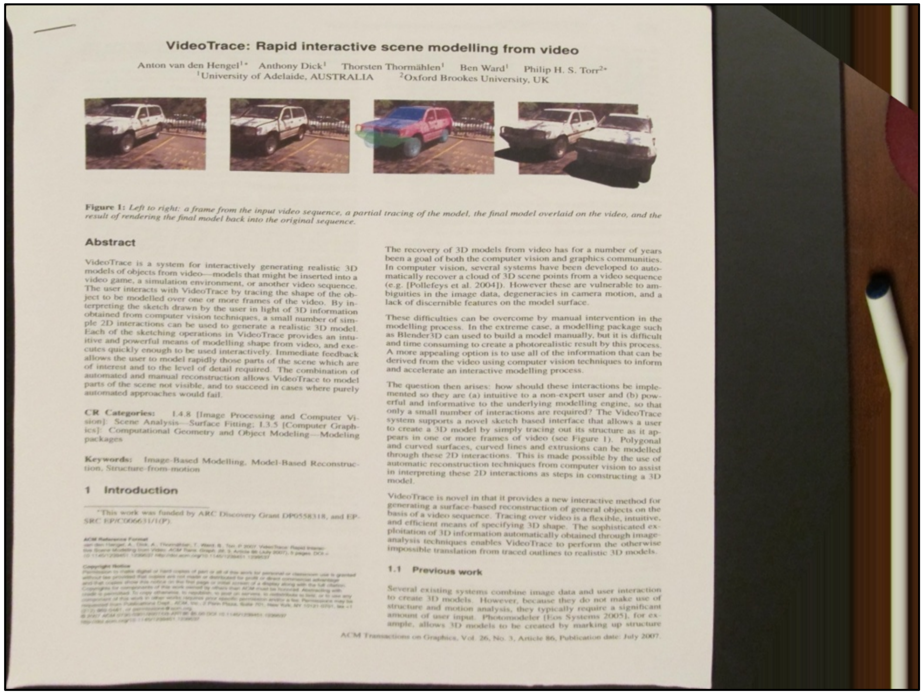
- **Construct homography H that maps the original (U,V) coordinates to rectified (U’,V’) coordinates (see Eqns 2.1, 2.2 & 2.3)**
  - **Verify that to extremely good approximation H(U,V,W)=(U’,V’,W’)**

Input image



- **Loop over U' & V' within new, rectified image whose pixel widths and heights equal those for UpsideDownPaper.jpg**
  - **For each (U',V') pair, find corresponding (U,V) via inverse homography H^{-1}**
  - **Extract the RGB value from UpsidedownPaper.jpg corresponding to the calculated (U,V)**
  - **Transfer RGB color to (U',V') location within rectified image**
- **Export rectified image as a JPEG file. Its text should be easy to read…**

Rectified output image



VideoTrace: Rapid interactive scene modelling from video

Anton van den Hengel[1*] Anthony Dick[1] Thorsten Thormählen[1] Ben Ward[1] Philip H. S. Torr[2*]
[1]University of Adelaide, AUSTRALIA [2]Oxford Brookes University, UK

Figure 1: *Left to right: a frame from the input video sequence, a partial tracing of the model, the final model overlaid on the video, and the result of rendering the final model back into the original sequence.*

Abstract

VideoTrace is a system for interactively generating realistic 3D models of objects from video—models that might be inserted into a video game, a simulation environment, or another video sequence. The user interacts with VideoTrace by tracing the shape of the object to be modelled over one or more frames of the video. By interpreting the sketch drawn by the user in light of 3D information obtained from computer vision techniques, a small number of simple 2D interactions can be used to generate a realistic 3D model. Each of the sketching operations in VideoTrace provides an intuitive and powerful means of modelling shape from video, and executes quickly enough to be used interactively. Immediate feedback allows the user to model rapidly those parts of the scene which are of interest and to the level of detail required. The combination of automated and manual reconstruction allows VideoTrace to model parts of the scene not visible, and to succeed in cases where purely automated approaches would fail.

CR Categories: I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Surface Fitting; I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Modeling packages

Keywords: Image-Based Modelling, Model-Based Reconstruction, Structure-from-motion

1 Introduction

*This work was funded by ARC Discovery Grant DP0558318, and EPSRC EP/C006631/1(P).

The recovery of 3D models from video has for a number of years been a goal of both the computer vision and graphics communities. In computer vision, several systems have been developed to automatically recover a cloud of 3D scene points from a video sequence (e.g. [Pollefeys et al. 2004]). However these are vulnerable to ambiguities in the image data, degeneracies in camera motion, and a lack of discernible features on the model surface.

These difficulties can be overcome by manual intervention in the modelling process. In the extreme case, a modelling package such as Blender3D can used to build a model manually, but it is difficult and time consuming to create a photorealistic result by this process. A more appealing option is to use all of the information that can be derived from the video using computer vision techniques to inform and accelerate an interactive modelling process.

The question then arises: how should these interactions be implemented so they are (a) intuitive to a non-expert user and (b) powerful and informative to the underlying modelling engine, so that only a small number of interactions are required? The VideoTrace system supports a novel sketch based interface that allows a user to create a 3D model by simply tracing out its structure as it appears in one or more frames of video (see Figure 1). Polygonal and curved surfaces, curved lines and extrusions can be modelled through these 2D interactions. This is made possible by the use of automatic reconstruction techniques from computer vision to assist in interpreting these 2D interactions as steps in constructing a 3D model.

VideoTrace is novel in that it provides a new interactive method for generating a surface-based reconstruction of general objects on the basis of a video sequence. Tracing over video is a flexible, intuitive, and efficient means of specifying 3D shape. The sophisticated exploitation of 3D information automatically obtained through image-analysis techniques enables VideoTrace to perform the otherwise impossible translation from traced outlines to realistic 3D models.

1.1 Previous work

Several existing systems combine image data and user interaction to create 3D models. However, because they do not make use of structure and motion analysis, they typically require a significant amount of user input. Photomodeler [Eos Systems 2005], for example, allows 3D models to be created by marking up structure

ACM Transactions on Graphics, Vol. 26, No. 3, Article 86, Publication date: July 2007.

Computer Lab 2B: Planar Mosaicing

- Download 3 photos shot via a rotating camera atop MIT's Green Building from http://web.mit.edu/alexv/Public/IAP_2013/Class2 . The files are called MITLeft.jpg, MITMiddle.jpg and MITRight.jpg

- Download text files from MITFeaturesLeftMiddle.txt and MITFeaturesRightMiddle.txt from http://web.mit.edu/alexv/Public/IAP_2013/Class2 . They contain manually selected feature tiepoints in Peter's 2D image plane coordinates

MITLeft image

MITMiddle image

MITRight image

Computer Lab 2B: Planar Mosaicing

- **First warp the left MIT image so that it aligns with the middle MIT photo (which we'll leave fixed).**
  - **Construct homography H that maps the left image's (U',V') coordinates to the middle image's (U,V) coordinates (see Eqns 2.1, 2.2 & 2.3)**
  - **Verify that to good approximation H(U',V',W') =(U,V,W)**

- **Use homography H to compute coordinates of left image's corners**

Lower left (Umin', Vmin')

Lower right (Umax',Vmin')

Upper right (Umax',Vmax')

Upper left (Umin', Vmax')

in terms of middle image's U & V variables

- **Compute bounding box B which encloses transformed left image's corners in middle image's UV coordinate space**

Computer Lab 2B: Planar Mosaicing

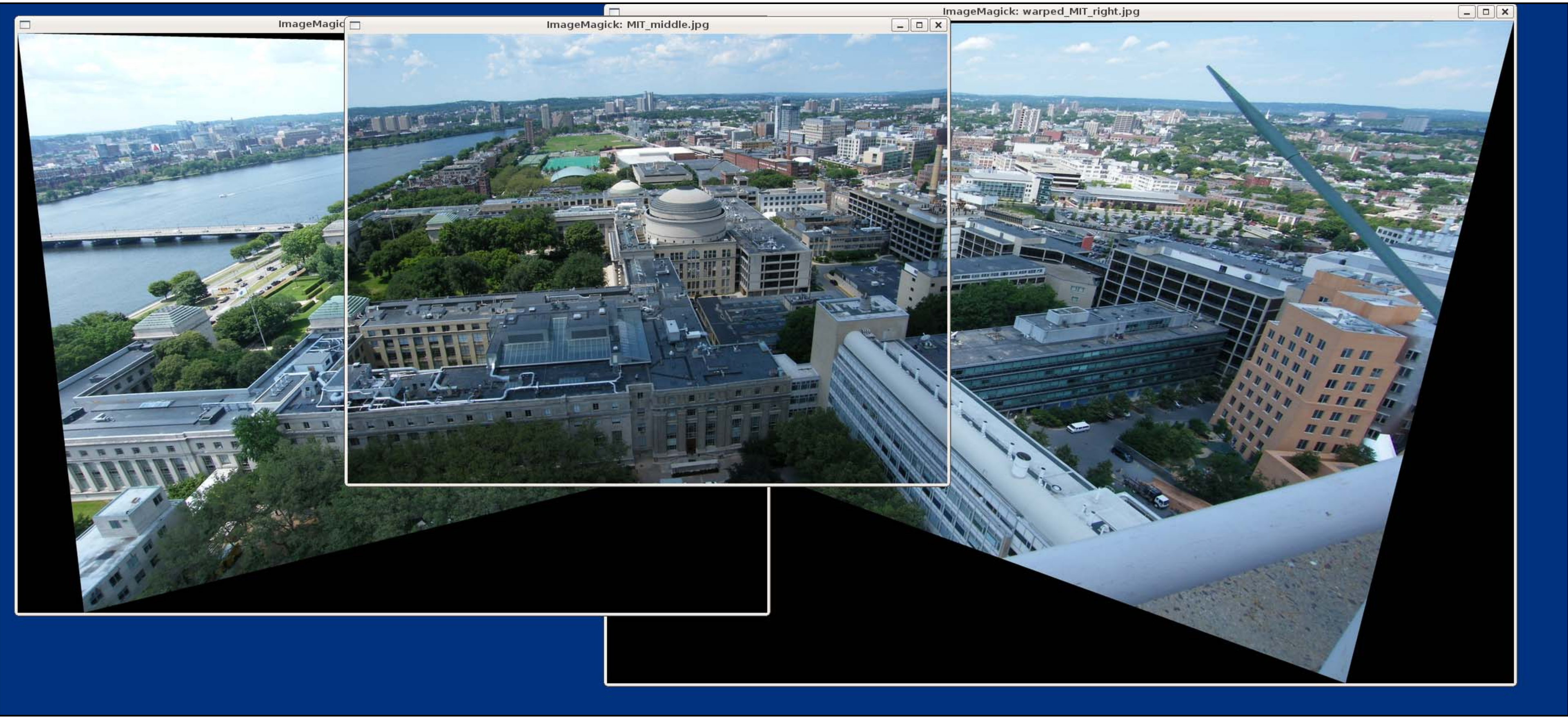
- Instantiate new blank image whose horizontal and vertical sizes are set by bounding box B
  - New image will hold warped version of left MIT photo

- Loop over U and V values within bounding box B
  - For each (U,V) pair, find corresponding (U',V') coordinates via inverse homography H^{-1}
  - Extract RGB value from MITLeft.jpg corresponding to calculated (U',V')
  - Transfer RGB color to (U,V) location within warped version of MITLeft.jpg

- Export warped version of MITLeft.jpg image as warpedMITLeft.jpg.
  - It should be easy to overlay warpedMITLeft.jpg onto MITMiddle.jpg to obtain a poor-man's mosaic

- Perform analogous homography warping for MITRight.jpg

Computer Lab 2B: Planar Mosaicing



Annotated References for Class 2

- **R. Hartley and A. Zisserman, "Multiple view geometry in computer vision (2nd edition)", Cambridge University Press, 2003.**
  - **Pedagogical material for today's class was drawn from chapters 2, 4 and 8**

- **A. Efros, "Homographies and Mosaics", 15-463: Computation photography notes, 2005.**
  - **There are several lectures about homographies and panoramas on the web. Here's one example which is quite good.**

- **P. Cho, S. Bae and F. Durand, "Image-based querying of urban knowledge databases", SPIE Proceedings, 2009.**
  - **The 3D static mosaic vs dynamic video results presented at the end of today's class come from this research paper.**