Calibrated Image Acquisition for Multi-view 3D Reconstruction

Sriram Kashyap M S

Guide: Prof. Sharat Chandran

Indian Institute of Technology, Bombay

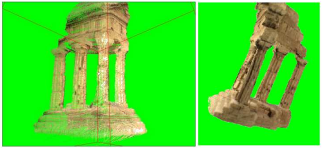
April 2009

Motivation

Given pictures of an object, how can I place the object in a virtual environment?

Motivation

Given pictures of an object, how can I place the object in a virtual environment?



SCATTERPLOT RENDERED

Traditional Graphics

- Render a 3D world on a 2D screen
- The world is authored using modeling tools, by artists

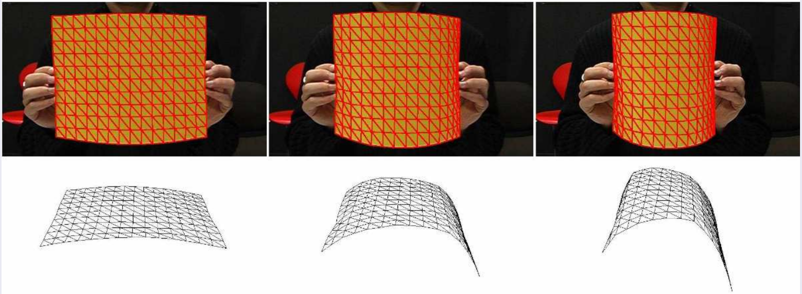
Traditional Graphics

- Render a 3D world on a 2D screen
- The world is authored using modeling tools, by artists

Going backwards: 2D to 3D

Going backwards: 2D to 3D

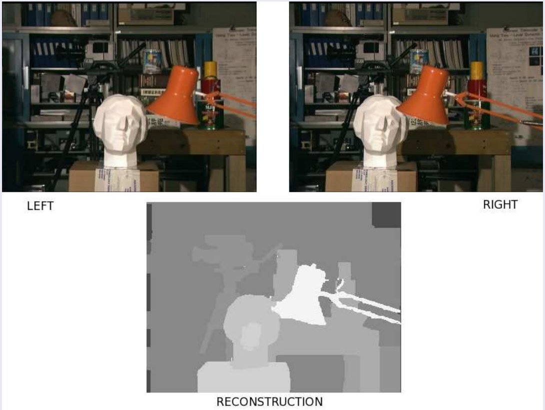
Monocular reconstruction



Salzmann et al. Local deformation models for monocular 3D shape recovery. CVPR 2008

Going backwards: 2D to 3D

Stereo Reconstruction



http://www.cs.washington.edu/homes/indria/project/CSE576finalproject/

Going backwards: 2D to 3D

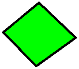
Multi-view Reconstruction

vision.middlebury.edu

Visual Hull

Visual hull is an approximate 3D structure obtained from reconstruction algorithms based on object silhouettes.

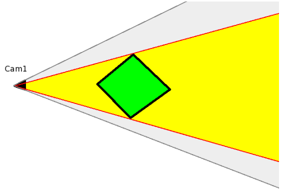
- Given a camera, a silhouette defines a back-projected generalized cone that contains the actual object
- Each camera gives us one such generalized cone
- The intersection of these cones is the Visual Hull



Visual Hull

Visual hull is an approximate 3D structure obtained from reconstruction algorithms based on object silhouettes.

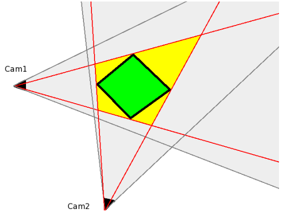
- Given a camera, a silhouette defines a back-projected generalized cone that contains the actual object
- Each camera gives us one such generalized cone
- The intersections of these cones is the Visual Hull



Visual Hull

Visual hull is an approximate 3D structure obtained from reconstruction algorithms based on object silhouettes.

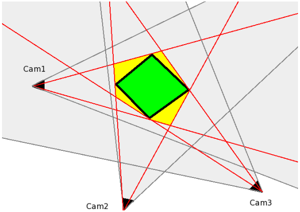
- Given a camera, a silhouette defines a back-projected generalized cone that contains the actual object
- Each camera gives us one such generalized cone
- The intersections of these cones is the Visual Hull



Visual Hull

Visual hull is an approximate 3D structure obtained from reconstruction algorithms based on object silhouettes.

- Given a camera, a silhouette defines a back-projected generalized cone that contains the actual object
- Each camera gives us one such generalized cone
- The intersections of these cones is the Visual Hull



Simulating multiple cameras

- Object is placed on a turntable
- Camera is fixed
- Camera captures frames as the table rotates

Issues in visual hull construction

- Image segmentation: Obtain silhouette information from images
- Camera calibration: Find the camera projection matrix for each camera

Segmentation

Background Subtraction

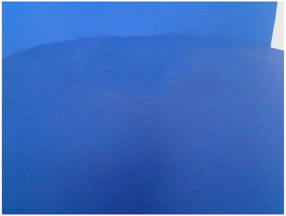
- Capture object and background images
- If background and object image pixels are similar, mark pixel as background
- Similarity tests performed in RGB and YCbCr color spaces

Segmentation

Background Subtraction

- Capture object and background images
- If background and object image pixels are similar, mark pixel as background
- Similarity tests performed in RGB and YCbCr color spaces

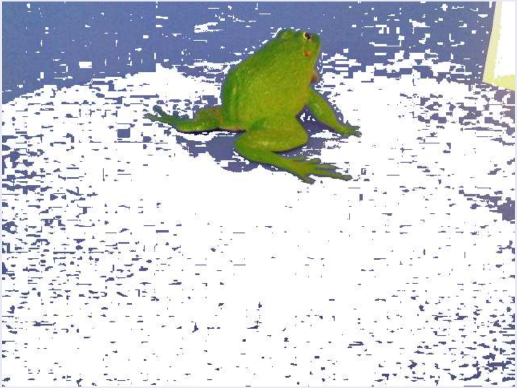
OBJECT



BACKGROUND

Segmentation

Segmented image



What went wrong?

- We are using a webcam
- Camera exposure, white balance, noise, compression artifacts
- What is the correct threshold value? This may vary from image to image

Adaptive thresholding

Can we provide more information to the system?

Adaptive thresholding

Can we provide more information to the system?

- Ratio of object pixels to total number of pixels remains within certain bounds
- Provide the expected upper bound for this ratio

Adaptive thresholding

Can we provide more information to the system?

- Ratio of object pixels to total number of pixels remains within certain bounds
- Provide the expected upper bound for this ratio
- Start with a low threshold value
- Increase the threshold till the ratio is below this upper bound

Adaptive thresholding

Segmented image (foreground ratio=0.07)

Camera Calibration

Calibration

Find the projection matrix corresponding to each camera view

Camera Calibration

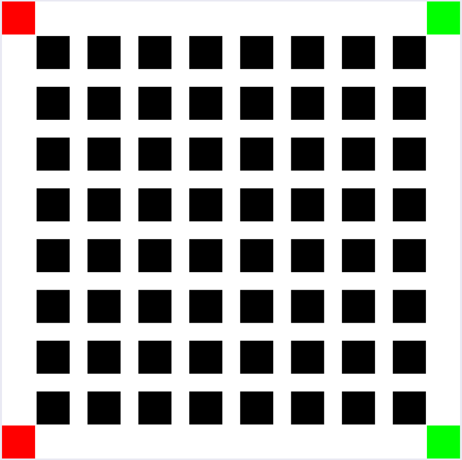
Calibration

Find the projection matrix corresponding to each camera view

- Find a set of 2D to 3D point correspondences
- Use existing tools to compute matrices from these correspondences

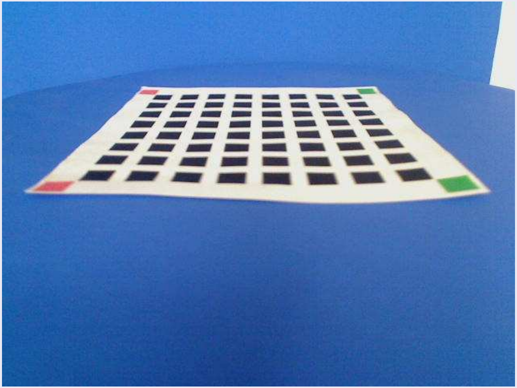
Point Correspondences

Calibration Pattern



Point Correspondences

Calibration Pattern



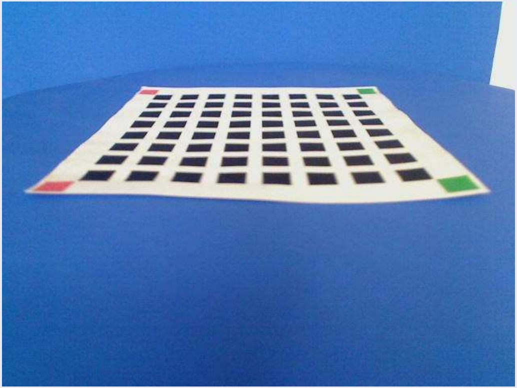
Point Correspondences

- Feature to find: Centers of black squares
- Centers are easier to find and more robust against errors

- Fix an absolute ordering of boxes
- Use colored boxes to find orientation
- Thresholding to locate the boxes
- Find centers of these boxes

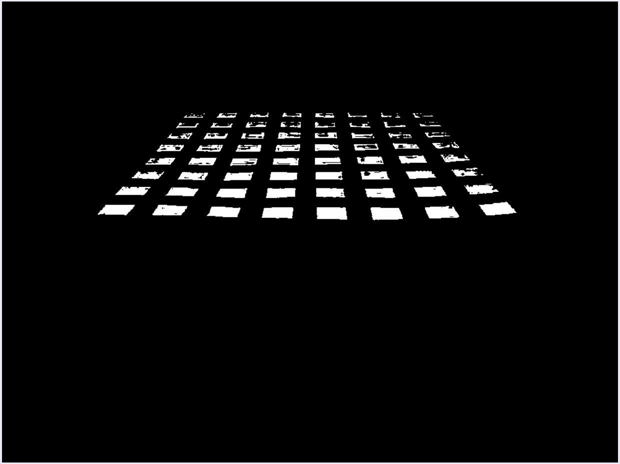
Example

Calibration Pattern



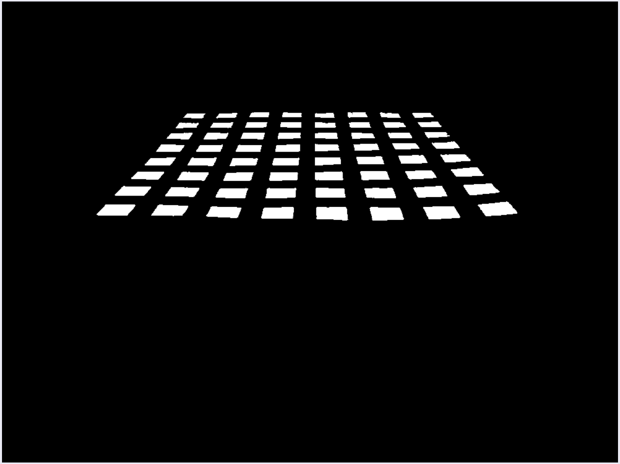
Example

Black threshold, first pass



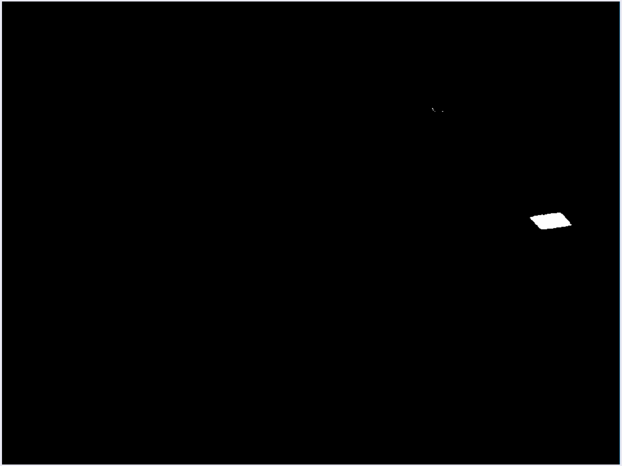
Example

Black threshold, second pass



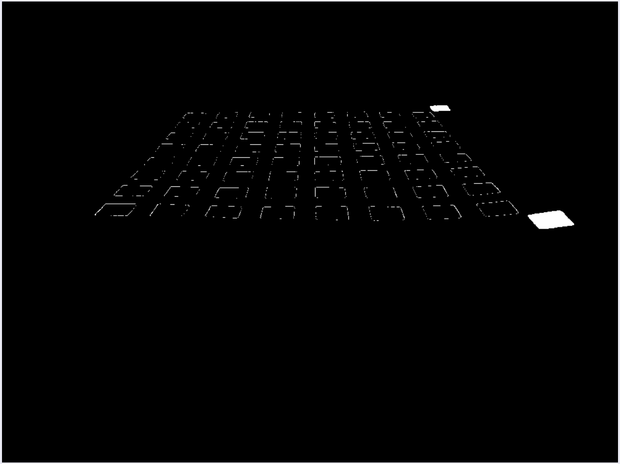
Example

Green threshold, first pass



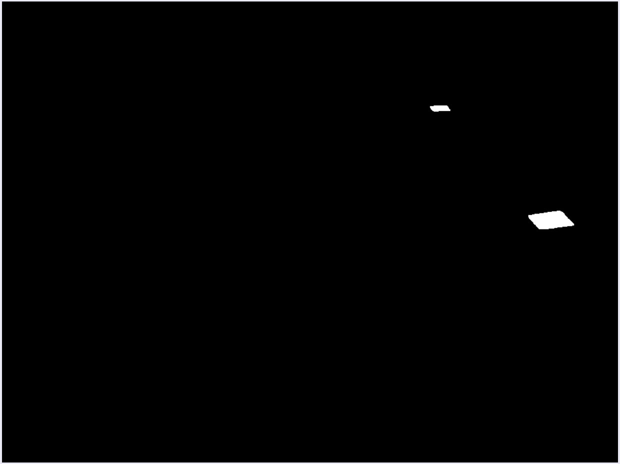
Example

Green threshold, second pass



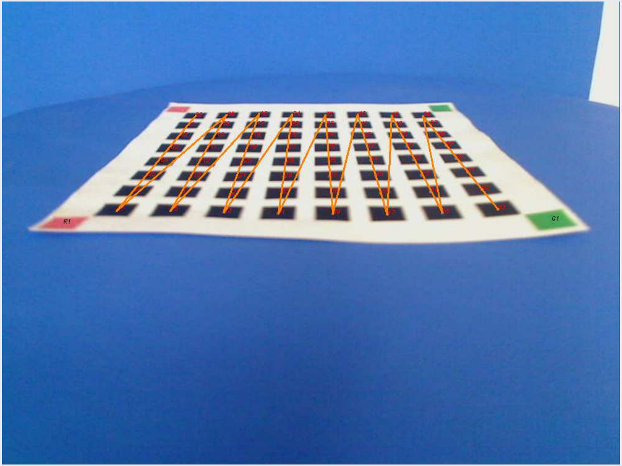
Example

Green threshold, final pass



Example

Labeled calibration pattern



Camera Calibration

- In some cases, the algorithm may explicitly fail(vision is uncertain)
- Discard such views automatically

Camera Calibration

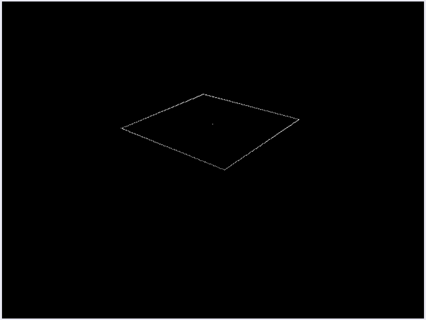
- In some cases, the algorithm may explicitly fail(vision is uncertain)
- Discard such views automatically

- In some cases, the algorithm may fail silently (returns an incorrect, but mathematically valid matrix)
- Cannot automatically discard such views, although tools can be written to help find bad views

Camera Calibration

Visualizing a camera matrix

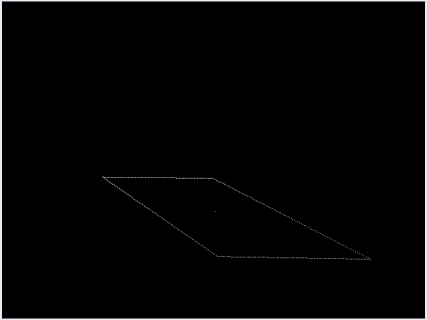
$$\begin{pmatrix} 216.201 & 634.052 & -44.0839 & 3922.02 \\ 36.6629 & -52.2481 & 394.727 & 3119.24 \\ -0.69624 & 0.685745 & -0.212142 & 12.13 \end{pmatrix}$$



Camera Calibration

Visualizing a camera matrix

$$\begin{pmatrix} 591.256 & -313.573 & -52.8837 & 3893.9 \\ -5.06723 & -59.6318 & -395.35 & 3148.67 \\ -0.0399772 & 0.974564 & -0.220515 & 11.9781 \end{pmatrix}$$

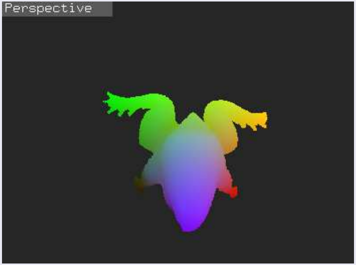


Results

Actual object

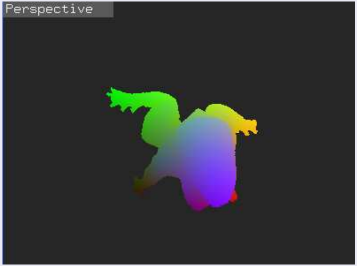
Results

Reconstruction



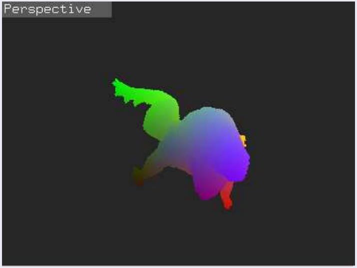
Results

Reconstruction



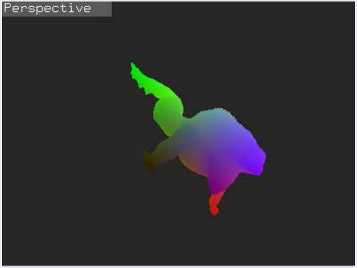
Results

Reconstruction



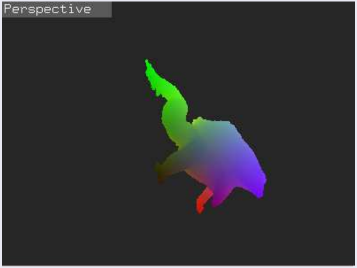
Results

Reconstruction



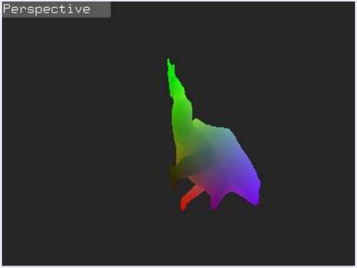
Results

Reconstruction



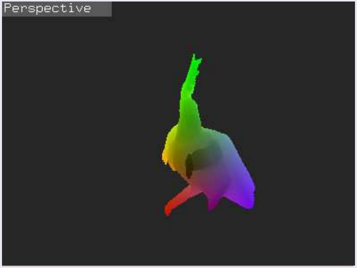
Results

Reconstruction



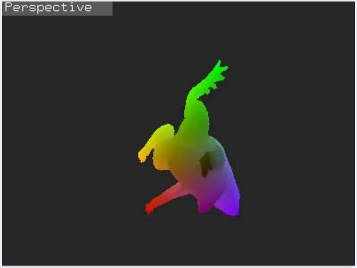
Results

Reconstruction



Results

Reconstruction



Future Work

1. Better calibration using optimization techniques
2. Textured rendering of visual hull
3. Image based Relighting support

References

1. OpenCV camera control and calibration: opencv.willowgarage.com/
2. Image based Animation: http://www.cse.iitb.ac.in/biswarup/projects/Motion/