Module: 3D Vision

Lecture 8: 3D reconstruction from multiple views.
Voxel-based methods.
Stratified methods.
Projective reconstruction.

Lecturer: Gloria Haro

Outline

- 3D reconstruction from multiple views
- Voxel-based methods
    - Shape from silhouette
    - Voxel coloring
    - Space carving
- Structure from motion.
- Stratified reconstruction.
- Projective reconstruction: Factorization method

3D reconstruction from multiple views

- Calibrated case: Multi-view stereo, Shape from X
- Non-calibrated case: Structure from motion

3D reconstruction from multiple views

- Calibrated case: Multi-view stereo, Shape from X
- Non-calibrated case: Structure from motion

Input data:
Images from different views, video sequence, images from databases, ...

3D reconstruction from multiple views

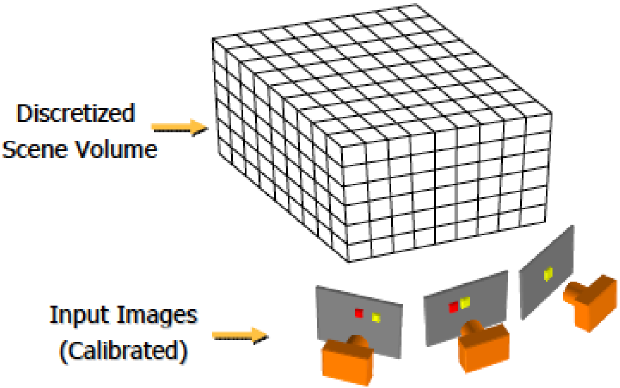
- Calibrated case: Multi-view stereo, Shape from X
- Non-calibrated case: Structure from motion

Input data:
Images from different views, video sequence, images from databases, ...

Representation of 3D shape:
Depth maps, voxels, meshes, point clouds, ...

Voxel-based methods

Problem statement



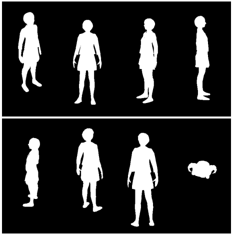
GOAL: Determine the **occupancy**, and eventually **color**, of voxels in **V**.

Image source: [S. Seitz]

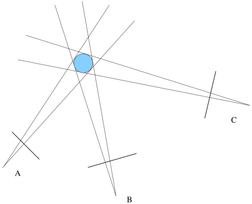
Shape from silhouette

Traditional approach: **Visual hull**

3D shape as the intersection of the back-projected silhouettes in the 3D space.



Silhouettes



Visual cones

Shape

Shape from silhouette

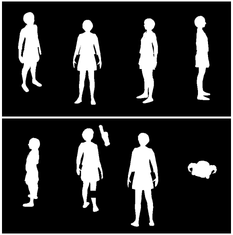
VISUAL HULL ALGORITHM

1. Iterate through all voxels in the volume
2. Project voxel to each image
3. If voxel projects inside silhouette in ALL views
   Mark the voxel as occupied

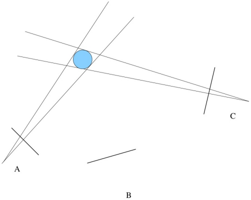
Shape from silhouette

Traditional approach: **Visual hull**

Fails in case of incomplete (inconsistent) silhouettes.
Robust methods exist.



Silhouettes



Visual cones

Shape

Shape from silhouette

OBSERVATION:
Visual hull methods **do not recover concavities**. Methods based on the photo-consistency of color images do recover them.

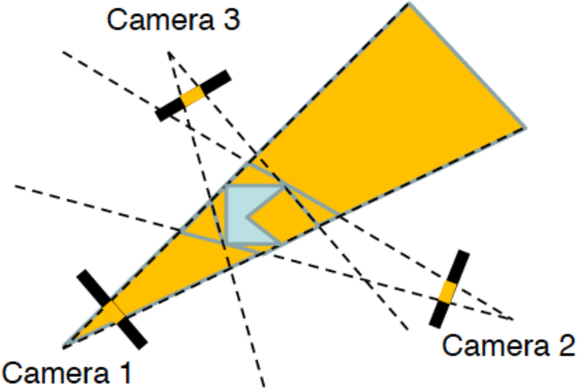
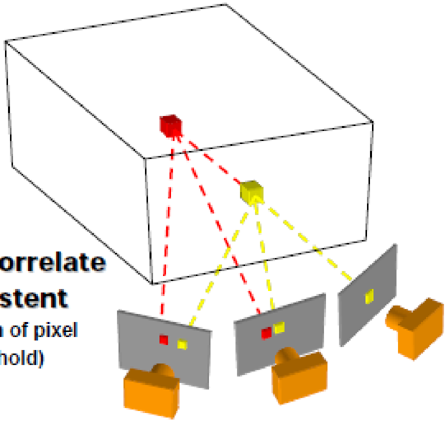


Image source: [A. Ladikos]

Voxel coloring



1. Choose voxel
2. Project and correlate
3. Color if consistent
   (standard deviation of pixel colors below threshold)

Image source: [S. Seitz]

Voxel coloring

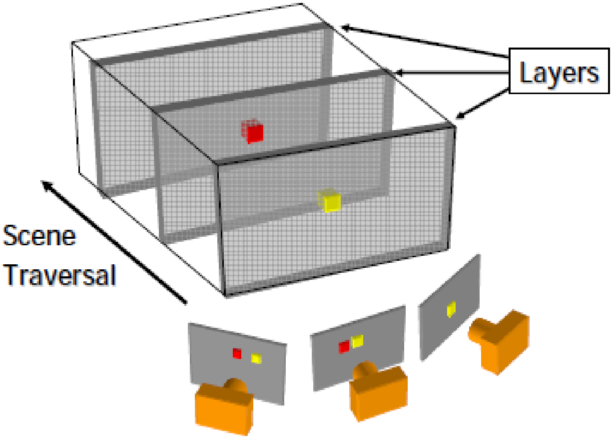


Image source: [S. Seitz]

Voxel coloring

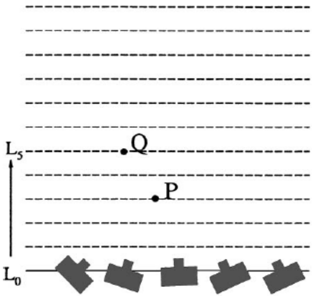
ALGORITHM

1. Iterate through the layers
2. Iterate through voxels in the layer
3. Project voxel to each image
4. Evaluate voxel photo-consistency
5. If photo-consistent
    - Color the voxel
    - Mark the image pixels (to determine voxel occlusion)

Voxel coloring

Constraint on the camera setting: if no scene point is contained within the convex hull of the camera centers the camera setting is compatible.

Examples of layered scene traversal



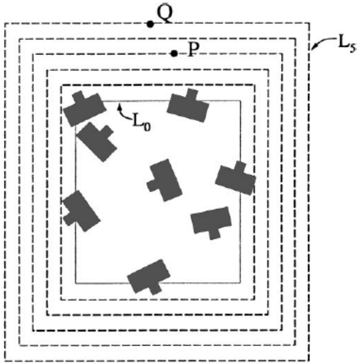


Image source: [S. Seitz]

Voxel coloring

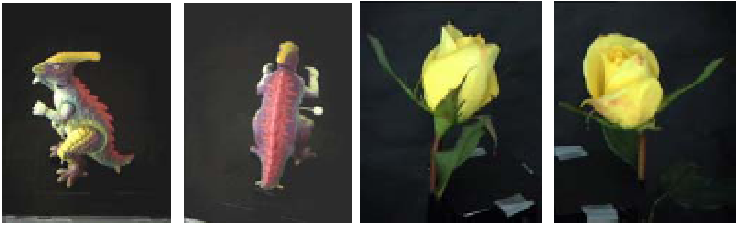


Image source: [S. Seitz]

Space carving

The 3D shape recovery from a set of images is generally **ill-posed**: there may be multiple shapes that are consistent with the same set of images.

Image source: [S. Lazebnik]

Space carving

The 3D shape recovery from a set of images is generally **ill-posed**: there may be multiple shapes that are consistent with the same set of images. $\rightarrow$ Some sort of constraint has to be imposed

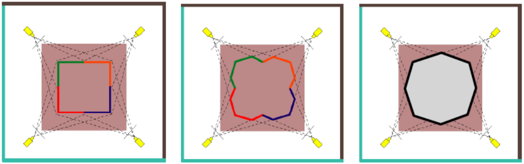
Image source: [S. Lazebnik]

Space carving

The 3D shape recovery from a set of images is generally **ill-posed**: there may be multiple shapes that are consistent with the same set of images. → Some sort of constraint has to be imposed

In particular:

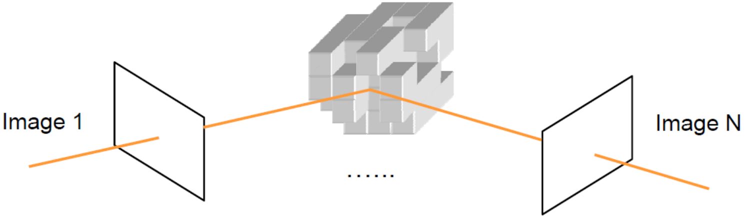
- The visual hull is defined as the maximal volume consistent with all the silhouettes.
- The voxel coloring uses the ordinal visibility constraint.
- Other multi-view stereo methods look for the shape with smoothest surface
- The space carving looks for the **photo hull**: the union of all photo-consistent shapes



True Scene Photo Hull Visual Hull

Image source: [S. Lazebnik]

Space carving

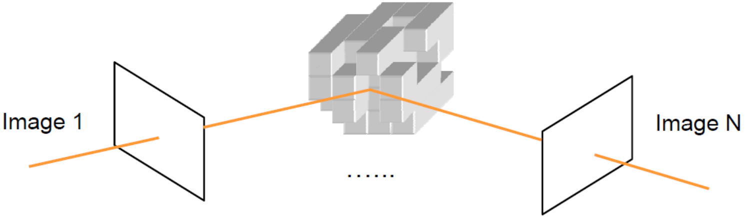


Space Carving Algorithm

- Initialize to a volume V containing the true scene
- Choose a voxel on the current surface
- Project to visible input images
- Carve if not photo-consistent
- Repeat until convergence

Image source: [S. Lazebnik]

Space carving



Space Carving Algorithm

- Initialize to a volume V containing the true scene
- Choose a voxel on the current surface
- Project to visible input images
- Carve if not photo-consistent
- Repeat until convergence

Needs to keep track of the scene visibility

Image source: [S. Lazebnik]

Space carving

Multi-view visibility ordering

Multi-view visibility orders do not exist in the general case BUT it is possible to define visibility orders that apply to a subset of the input cameras.

Image source: [Kutulakos and Seitz 2000]

Space carving

Multi-view visibility ordering

Multi-view visibility orders do not exist in the general case BUT it is possible to define visibility orders that apply to a subset of the input cameras.

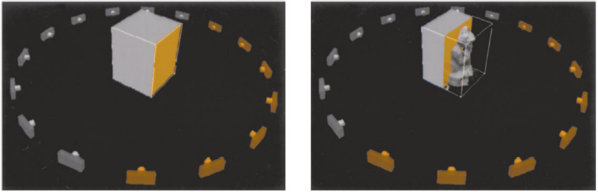
$\rightarrow$ No constraints on the camera setting

Image source: [Kutulakos and Seitz 2000]

Space carving

Multi-view visibility ordering

Multi-view visibility orders do not exist in the general case BUT it is possible to define visibility orders that apply to a subset of the input cameras.
$\rightarrow$ No constraints on the camera setting



Multi-sweep space carving algorithm

Performs 6 sweeps through the volume, corresponding to the 6 principle directions (increasing and decreasing X, Y, and Z)

Image source: [Kutulakos and Seitz 2000]

Multi-view stereo methods

Volumetric methods based on an energy minimization.

They use a photo-consistency measure and sometimes combine the silhouettes information.

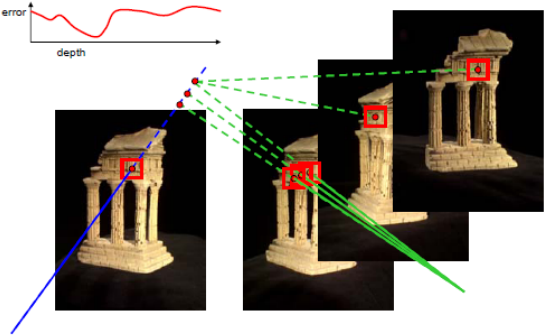
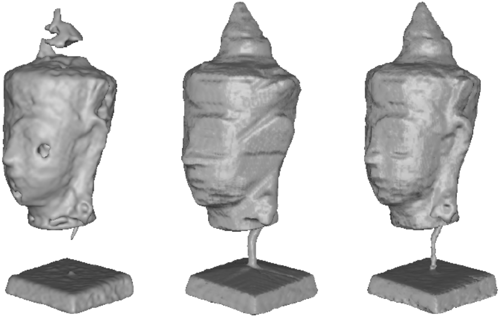


Image source: [S. Seitz]

Multi-view stereo methods



stereo reconstr. | silhouette reconstr. | stereo & silhouettes

Image source: [Cremers and Kolev 2011]

Outline

- 3D reconstruction from multiple views
- Voxel-based methods
  - Shape from silhouette
  - Voxel coloring
  - Space carving
- Structure from motion.
- Stratified reconstruction.
- Projective reconstruction: Factorization method

Structure from motion

Problem statement
Given a set of uncalibrated images and a set of image correspondences, compute the 3D points and the position, orientation, and calibration of the cameras (P matrices).

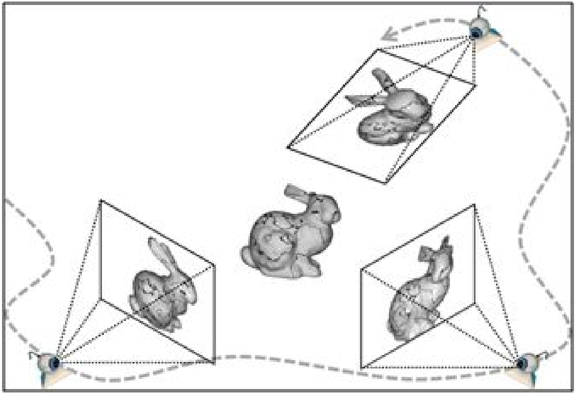


Image source: I. Mitsugami

Structure from motion

Image source: [Snavely et al. 2007]

Structure from motion

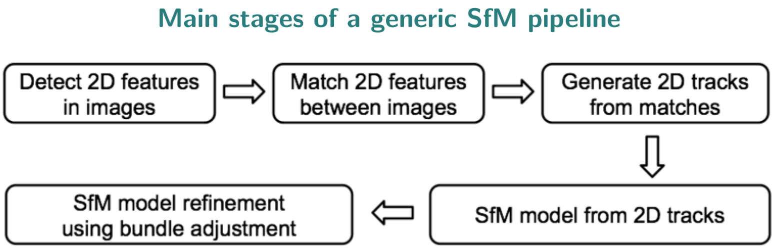
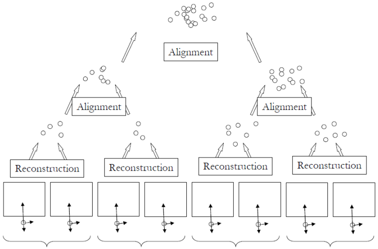


Image source: [Furukawa and Hernández 2013]

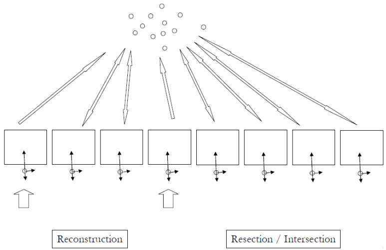
Structure from motion

Different approaches:

- Batch
  - Direct: all views at once
  - Hierarchical
- Sequential



Hierarchical



Sequential

Images source: A. Bartoli

Outline

- 3D reconstruction from multiple views
- Voxel-based methods
  - Shape from silhouette
  - Voxel coloring
  - Space carving
- Structure from motion.
- Stratified reconstruction.
- Projective reconstruction: Factorization method

Projective ambiguity

Given a set of uncalibrated images and a set of image correspondences, compute the 3D points and the position, orientation, and calibration of the cameras (P matrices).

Image calibration contains an inherent **projective ambiguity**

$$\mathbf{x} = P\mathbf{X}, \text{ but also } \mathbf{x} = PH^{-1}H\mathbf{X} = \widehat{P}\widehat{\mathbf{X}}$$

Projective ambiguity

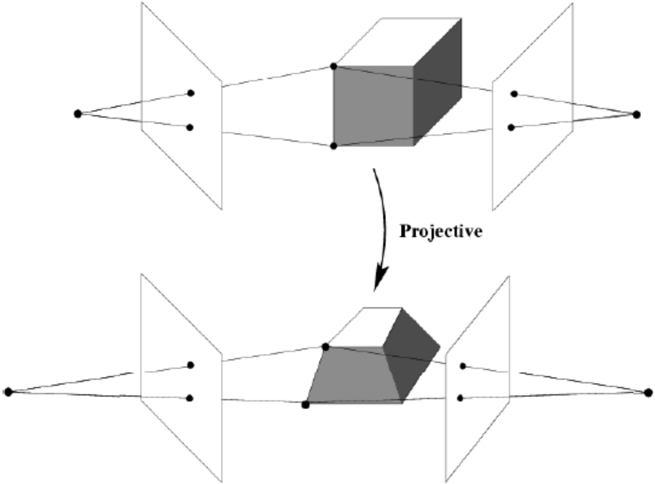


Image source: [Hartley and Zisserman 2004]

Projective reconstruction

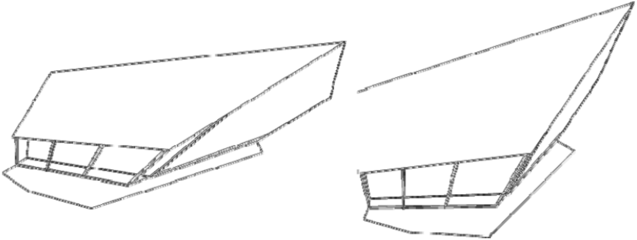


Image source: [Hartley and Zisserman 2004]

Stratified reconstruction

A solution to the projective ambiguity problem is the **stratified reconstruction**.

The main steps are:

1. Estimate a projective reconstruction
2. Upgrade the previous recons. to an affine reconstruction (OPTIONAL)
3. Upgrade the previous recons. to a metric reconstruction

Affine ambiguity

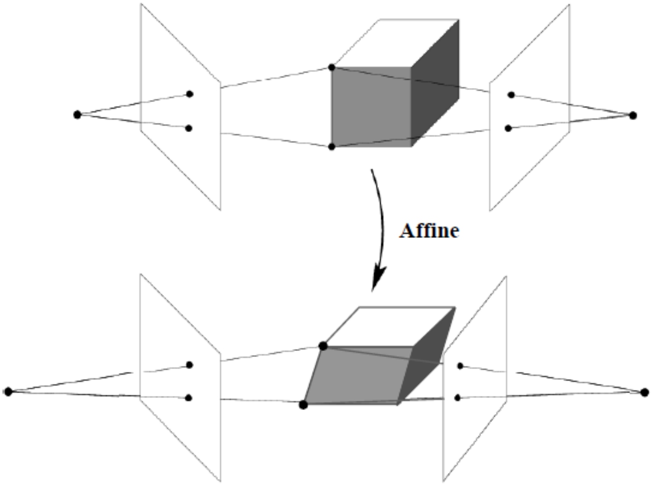


Image source: [Hartley and Zisserman 2004]

Affine reconstruction

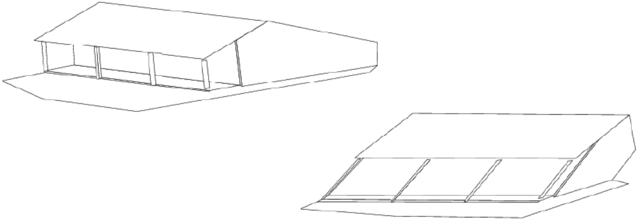


Image source: [Hartley and Zisserman 2004]

Similarity (or metric) ambiguity

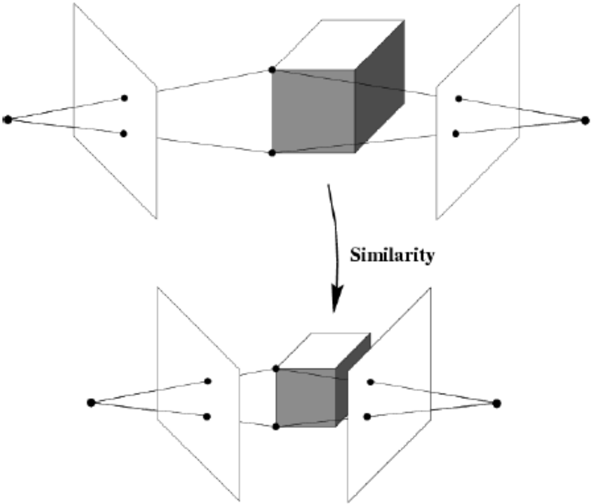


Image source: [Hartley and Zisserman 2004]

Metric reconstruction

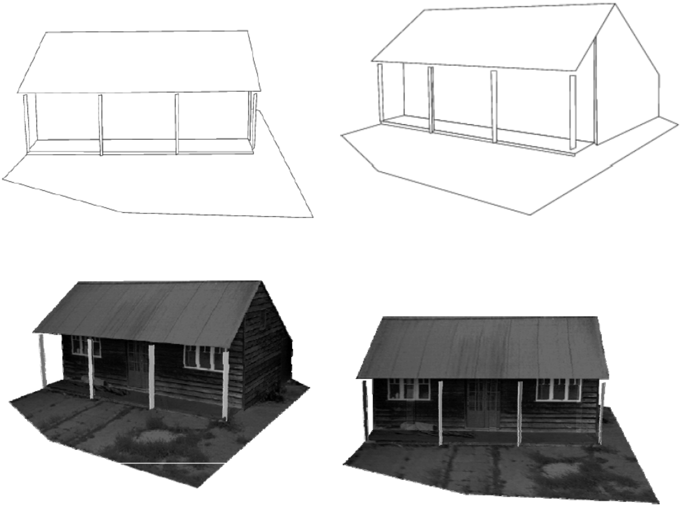


Image source: [Hartley and Zisserman 2004]

Outline

- 3D reconstruction from multiple views
- Voxel-based methods
  - Shape from silhouette
  - Voxel coloring
  - Space carving
- Stratified reconstruction
- Projective reconstruction: Factorization method

Projective reconstruction

Particular case of 2 views

If we estimate F we can extract two possible camera matrices:

$$P = [I \mid 0]$$

$$P' = [SF \mid e']$$

where S is any skew-symmetric matrix (such that P' has rank 3).
A good choice is: $S = [e']_x$.
The epipole may be computed from $e'^T F = 0$

Factorization method

Projective reconstruction method for 2 or more views
[P. Sturm and B. Triggs 1996]

Projective equations:

$$\mathbf{x}_j^i \equiv P^i \mathbf{X}_j \qquad \longrightarrow \qquad \lambda_j^i \mathbf{x}_j^i = P^i \mathbf{X}_j$$

where $j = 1, \dots n$ denote the points, and
$i = 1, \dots m$ denote the images (views)

Factorization method

Projective reconstruction method for 2 or more views
[P. Sturm and B. Triggs 1996]

Projective equations:

$$\mathbf{x}_j^i \equiv P^i \mathbf{X}_j \qquad \longrightarrow \qquad \lambda_j^i \mathbf{x}_j^i = P^i \mathbf{X}_j$$

where $j = 1, \ldots n$ denote the points, and
$i = 1, \ldots m$ denote the images (views)

Collect all projective eq's into a matrix equation:

$$\begin{bmatrix} \lambda_1^1 \mathbf{x}_1^1 & \lambda_2^1 \mathbf{x}_2^1 & \ldots & \lambda_n^1 \mathbf{x}_n^1 \\ \lambda_1^2 \mathbf{x}_1^2 & \lambda_2^2 \mathbf{x}_2^2 & \ldots & \lambda_n^2 \mathbf{x}_n^2 \\ \ldots & \ldots & \ldots & \ldots \\ \lambda_1^m \mathbf{x}_1^m & \lambda_2^m \mathbf{x}_2^m & \ldots & \lambda_n^m \mathbf{x}_n^m \end{bmatrix} = \begin{bmatrix} P^1 \\ P^2 \\ \ldots \\ P^m \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \ldots & \mathbf{X}_n \end{bmatrix}$$

λ_j^i are unknown scalar factors, called **projective depths**

Factorization method

Projective reconstruction method for 2 or more views
[P. Sturm and B. Triggs 1996]

Projective equations:

$$\mathbf{x}_j^i \equiv P^i \mathbf{X}_j \qquad \longrightarrow \qquad \lambda_j^i \mathbf{x}_j^i = P^i \mathbf{X}_j$$

where $j = 1, \ldots n$ denote the points, and
$i = 1, \ldots m$ denote the images (views)

Collect all projective eq's into a matrix equation:

$$\begin{bmatrix} \lambda_1^1 \mathbf{x}_1^1 & \lambda_2^1 \mathbf{x}_2^1 & \ldots & \lambda_n^1 \mathbf{x}_n^1 \\ \lambda_1^2 \mathbf{x}_1^2 & \lambda_2^2 \mathbf{x}_2^2 & \ldots & \lambda_n^2 \mathbf{x}_n^2 \\ \ldots & \ldots & \ldots & \ldots \\ \lambda_1^m \mathbf{x}_1^m & \lambda_2^m \mathbf{x}_2^m & \ldots & \lambda_n^m \mathbf{x}_n^m \end{bmatrix} = \begin{bmatrix} P^1 \\ P^2 \\ \ldots \\ P^m \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \ldots & \mathbf{X}_n \end{bmatrix}$$

λ_j^i are unknown scalar factors, called **projective depths**

Requires a set of points visible in all views!

Factorization method

We have

$$\underbrace{M}_{3m \times n} = \underbrace{\begin{bmatrix} P^1 \\ P^2 \\ \dots \\ P^m \end{bmatrix}}_{3m \times 4} \underbrace{\begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \dots & \mathbf{X}_n \end{bmatrix}}_{4 \times n}$$

$$M = \mathcal{P}_M \mathcal{X}_M$$

M is called the **measurement matrix**.

M has at most rank 4 $\rightarrow$ this suggests a factorization algorithm based on the SVD

Factorization method

$$M = \underbrace{U}_{3m \times n} \underbrace{D}_{n \times n} \underbrace{V^T}_{n \times n}$$

D should ideally have only four non-zero elements (M rank 4).

Factorization method

$$M = \underbrace{U}_{3m \times n} \underbrace{D}_{n \times n} \underbrace{V^T}_{n \times n}$$

D should ideally have only four non-zero elements (M rank 4).

We define D_4 as the $n \times 4$ submatrix of D.

Factorization method

$$M = \underbrace{U}_{3m \times n} \underbrace{D}_{n \times n} \underbrace{V^T}_{n \times n}$$

D should ideally have only four non-zero elements (M rank 4).

We define D_4 as the $n \times 4$ submatrix of D.

We write:

$$M = \underbrace{UD_4}_{3m \times 4} \underbrace{V_4^T}_{4 \times n}$$

where

$$UD_4 = \begin{bmatrix} P^1 \\ P^2 \\ \dots \\ P^m \end{bmatrix} \qquad V_4^T = \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \dots & \mathbf{X}_n \end{bmatrix}$$

Note: the factorization is not unique

Factorization method

Normalization of the data with a similarity transformation so that transformed points have zero mean and average distance from the origin of $\sqrt{2}$.

Apply the following similarity transformation to each image i:

$$H_s^i = \begin{pmatrix} s^i & 0 & -s^i c_x^i \\ 0 & s^i & -s^i c_y^i \\ 0 & 0 & 1 \end{pmatrix}$$

where centroid $c^i = (c_x^i, c_y^i)$, and $s^i = \frac{\sqrt{2}}{\text{mean dist to } c^i}$.

Limitation: the 3D points must be visible in all the views.

Factorization method

ALGORITHM

1. Determine a subset of scene points and cameras so that the measurement matrix is completely determined.
2. Normalize the set of points in each image (similarity transf. H_s).
3. Initialize all λ_j^i ($= 1$ or better initialization).
4. Alternate rescaling the rows of the depth matrix Λ (formed by λ_j^i) to have unit norm and the columns of Λ to have unit norm until Λ stops changing significantly (usually two loops).
5. Build the measurement matrix M.
6. Determine the SVD of $M = UDV^T$.
7. Let $\mathcal{P}_M = UD_4$ and $\mathcal{X}_M = V_4^T$.
8. If $\sum_i \sum_j d(x_j^i, P^i \mathbf{X}_j)^2$ converges then stop.
   Otherwise let $\lambda_j^i = (P^i \mathbf{X}_j)_3$ and go to Step 4.
9. Unnormalize the camera matrices $(H_s^i)^{-1} P^i$.
10. (Triangulate and resection the non-nucleus scene points and cameras).

References

[Hartley and Zisserman 2004] R.I. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2004.

[Hernández and Furukawa 2013] C. Hernández and Y. Furukawa, Multi-view stereo: A tutorial. Foundations and Trends® in Computer Graphics and Vision, 9 (1-2), 2013.

[K. Kutulakos and S. Seitz 2000] K. Kutulakos and S. Seitz, A theory of shape by space carving. Int. Journal of Computer Vision, 38(3) 2000.

[A. Laurentini 1994], The visual hull concept for silhouette-based image understanding, IEEE Trans. Pattern Analysis and Machine Learning, 16(2), 1994.

[S. Seitz and C. Dyer 1999] S. Seitz, and C. Dyer, Photorealistic Scene Reconstruction by Voxel Coloring. Int. Journal of Computer Vision, 35,(2) 1999.

[Snavely et al. 2007] N. Snavely, S. M. Seitz, R. Szeliski. Modeling the World from Internet Photo Collections. Int. Journal of Computer Vision, 2007.

[P. Sturm and B. Triggs 1996] P. Sturm and B. Triggs, A factorization based algorithm for multi-image projective structure and motion, European Conference on Computer Vision, 1996.