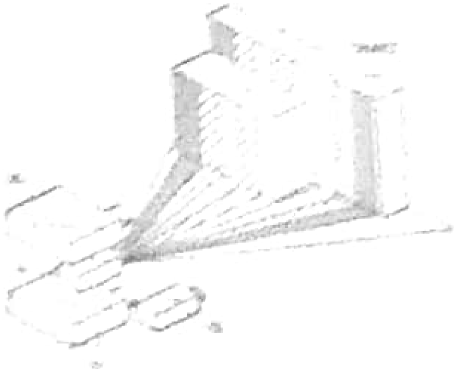
# Structured Lighting

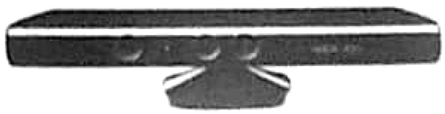
- Finding correspondences is hard by itself
- Can we help it by projecting patterns onto the world?
- Structured Lights!
- Lightstrip range finders, etc.
- Combination of sinusoids sometimes to get dense matches
- Active vision, as it changes the appearance
- The light projected need not be in the visible spectrum



# Xbox Kinect

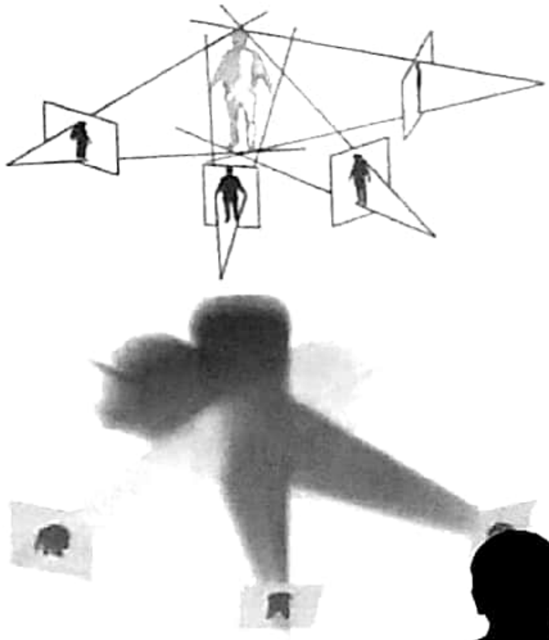
IR-based range sensor for Xbox

- Aligned depth and RGB images at 640 × 480
- Original goal: Interact with games in full 3D
- Computer vision happy with real-time depth and image
  - Games, HCI, etc
  - Action recognition
  - Image based modelling of dynamic scenes
- Fastest selling electronic appliance ever!!
- Other products that use PrimeSense sensor



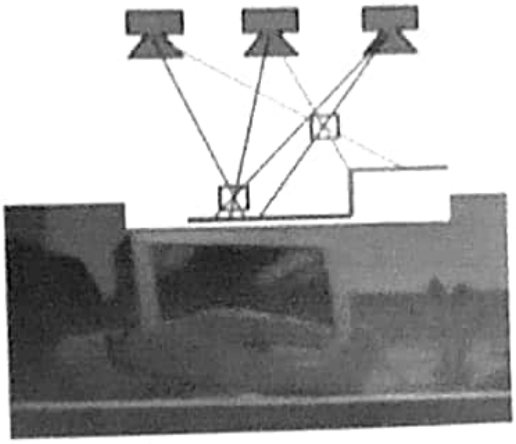
# Visual Hull

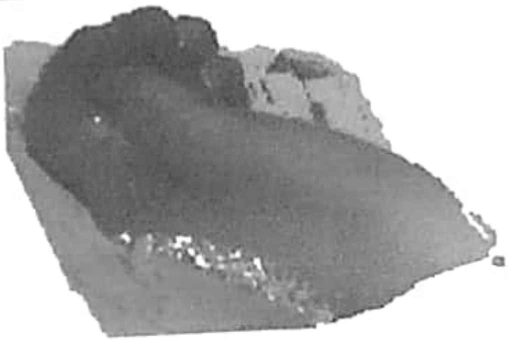
- Object silhouette represents a generalized cone with the camera centre as the apex
- Intersect these cones for multiple views in the 3D space
- Visual Hull, like convex hull
- Cannot get fine details like concavities
- Gives a very good, approximate shape, without scene modification!



# Space Carving

- Reason directly in a volumetric voxel space
- If a voxel is filled, it projects to similar colours in all cameras
- If a voxel is empty, its projections will have different appearances
- Colour consistency: filled or empty?
- Assume all filled initially; carve out empty ones by going over the images, guessing visibility, etc.



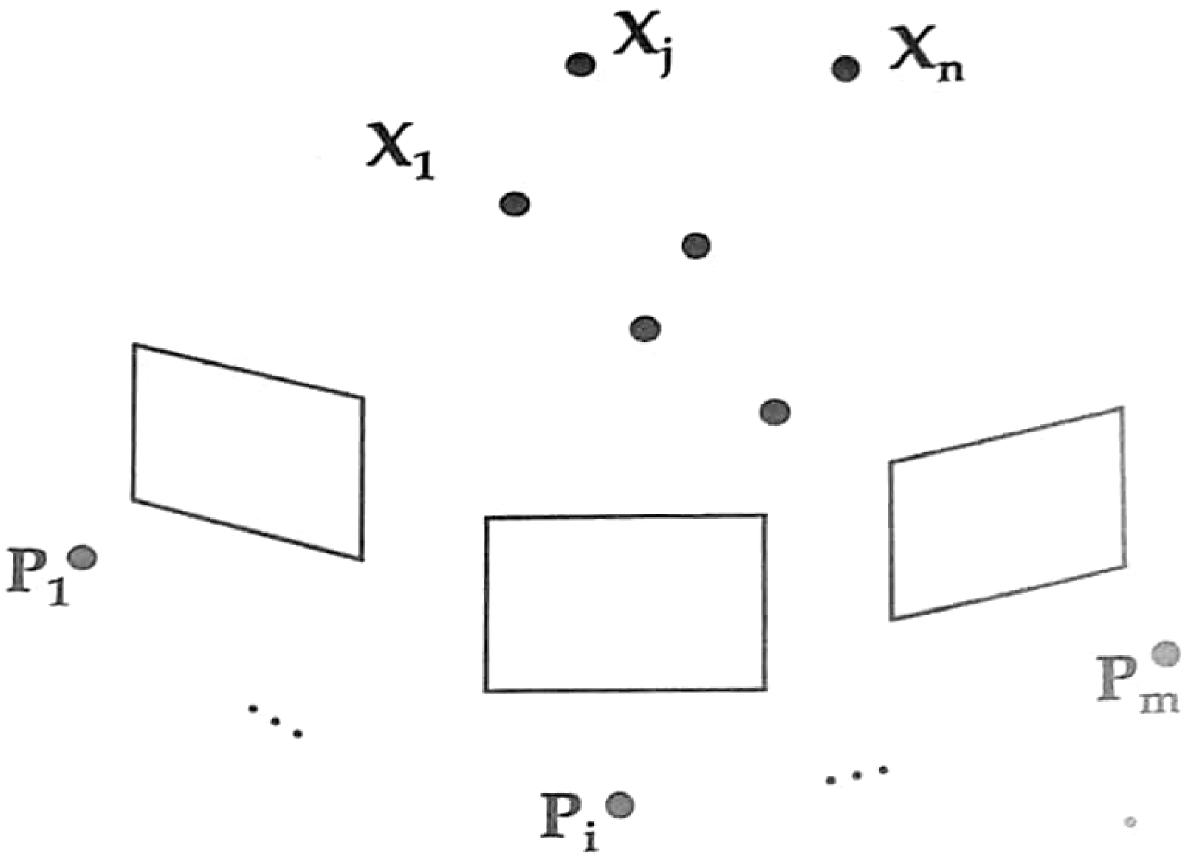


# CSE578: Computer Vision

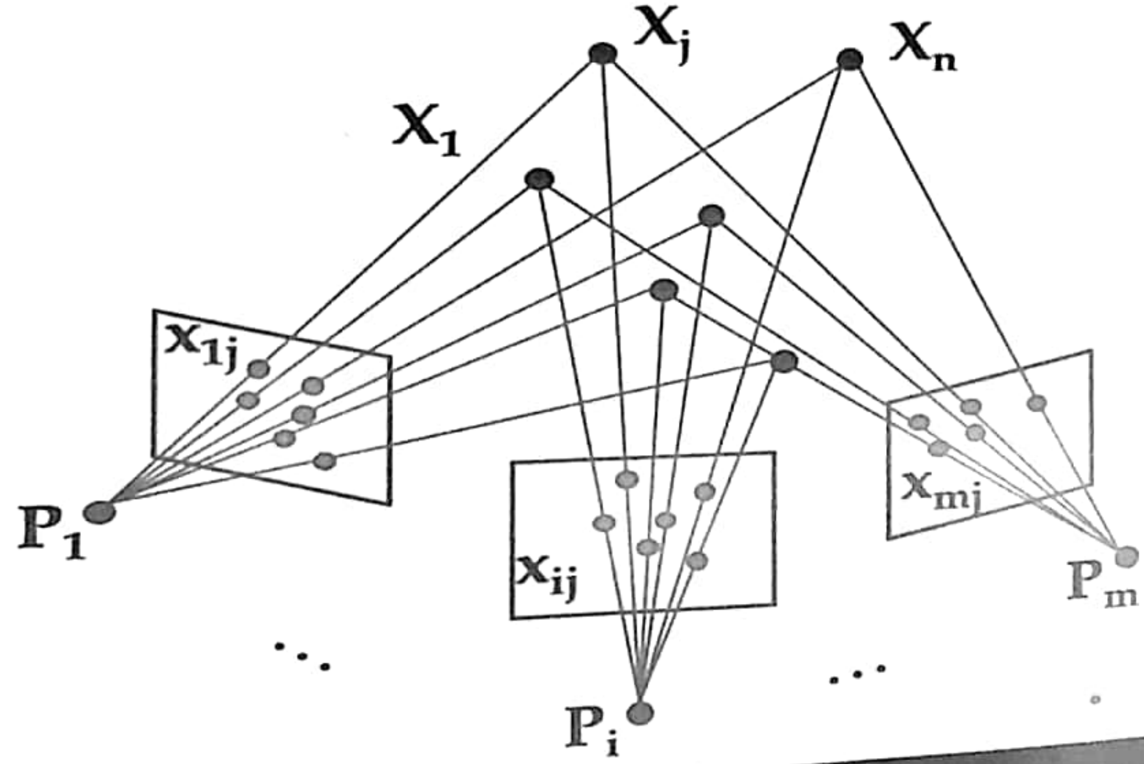
## 3D Reconstruction:
## Structure from Motion using Bundle Adjustment

Anoop M. Namboodiri
Center for Visual Information Technology
IIIT Hyderabad, INDIA

# Points in Multiple Views

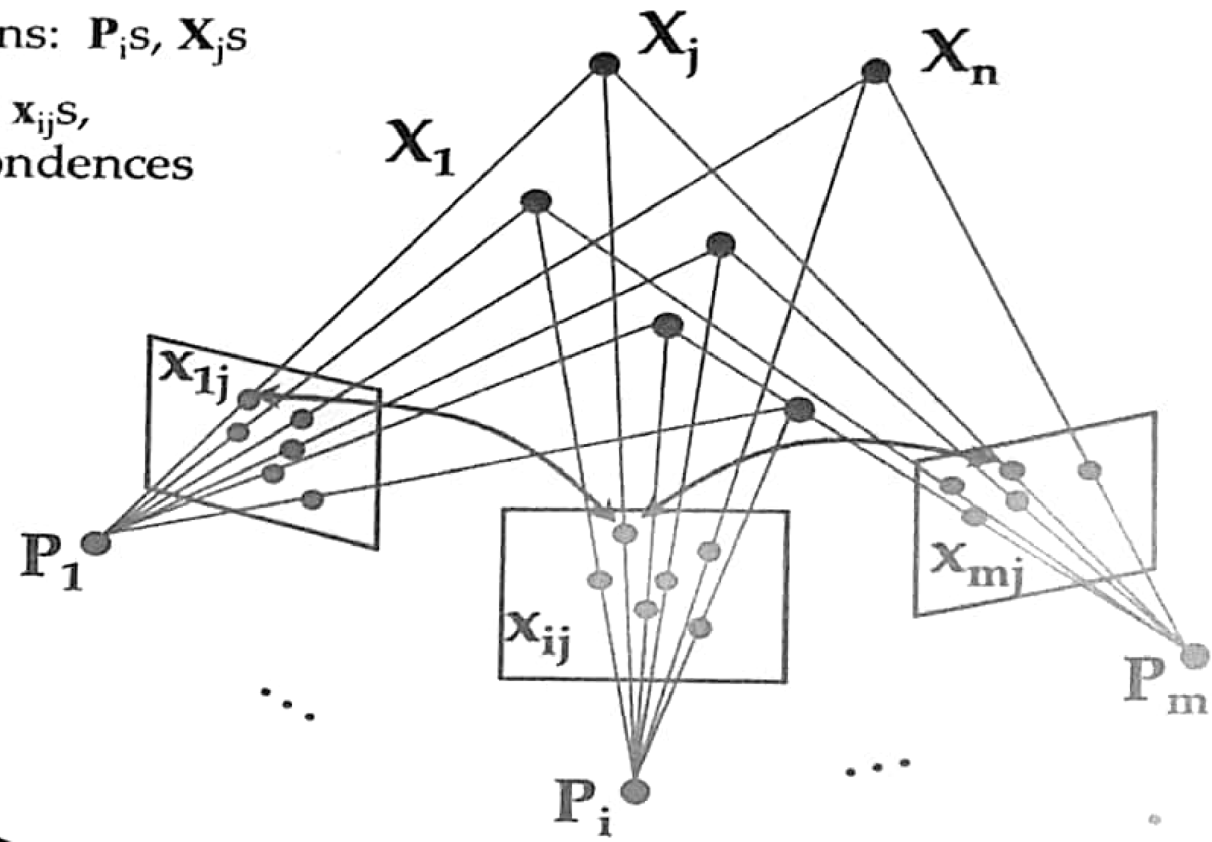


# Points in Multiple Views



# Points in Multiple Views

- Unknowns: $\mathbf{P}_i$s, $\mathbf{X}_j$s
- Knowns: $\mathbf{x}_{ij}$s, Correspondences



# Structure from Motion

- $m$ cameras and $n$ points, with correspondences
- Unknown: $m$ matrices $\mathbf{P}_i$ and $n$ coordinates, $\mathbf{X}_j$
- We have: $x_{ij} = \mathbf{P}_i\mathbf{X}_j$ , $1 \leq i \leq m$, $1 \leq j \leq n$
- $2mn$ equations in total (2 for each visible point)
- Can be solved if $2mn > 11m + 3n$
- However, under projective, $\mathbf{PX} = (\mathbf{PQ})(\mathbf{Q}^{-1}\mathbf{X})$, a projective ambiguity will remain
- Projective structure if $2mn > 11m + 3n - 15$
- Affine structure if $2mn > 11m + 3n - 12$
- Metric structure if $2mn > 11m + 3n - 7$
- Affine/Metric structure only by enforcing affine/metric constraints

# Structure from Motion

- $m$ cameras and $n$ points, with correspondences
- Unknown: $m$ matrices $\mathbf{P}_i$ and $n$ coordinates, $\mathbf{X}_j$
- We have: $x_{ij} = \mathbf{P}_i\mathbf{X}_j$ , $1 \leq i \leq m$, $1 \leq j \leq n$
- $2mn$ equations in total (2 for each visible point)
- Can be solved if $2mn > 11m + 3n$
- However, under projective, $\mathbf{PX} = (\mathbf{PQ})(\mathbf{Q}^{-1}\mathbf{X})$, a projective ambiguity will remain
- Projective structure if $2mn > 11m + 3n - 15$
- Affine structure if $2mn > 11m + 3n - 12$
- Metric structure if $2mn > 11m + 3n - 7$
- Affine/Metric structure only by enforcing affi metric constraints

# Bundle Adjustment

- Given $m$ views of $n$ 3D points, with unknown $\mathbf{P}_i$ and $\mathbf{X}_j$. Ideally, $\mathbf{x}_{ij} = \mathbf{P}_i\mathbf{X}_j$
- Minimize the re-projection error over all cameras/views:

$$\min_{P_i X_j} \sum_{i=1}^{m} \sum_{i=1}^{n} dist(X_{ij}, P_i X_j)^2$$

- A non-linear optimization problem. Can be solved using the Levenberg-Marquardt procedure directly.
- Called bundle adjustment. Known to photogrammetry community for a long time
- Needs good initialization as the complex non-linear optimization problem can get stuck in local minima

# Bundle Adjustment

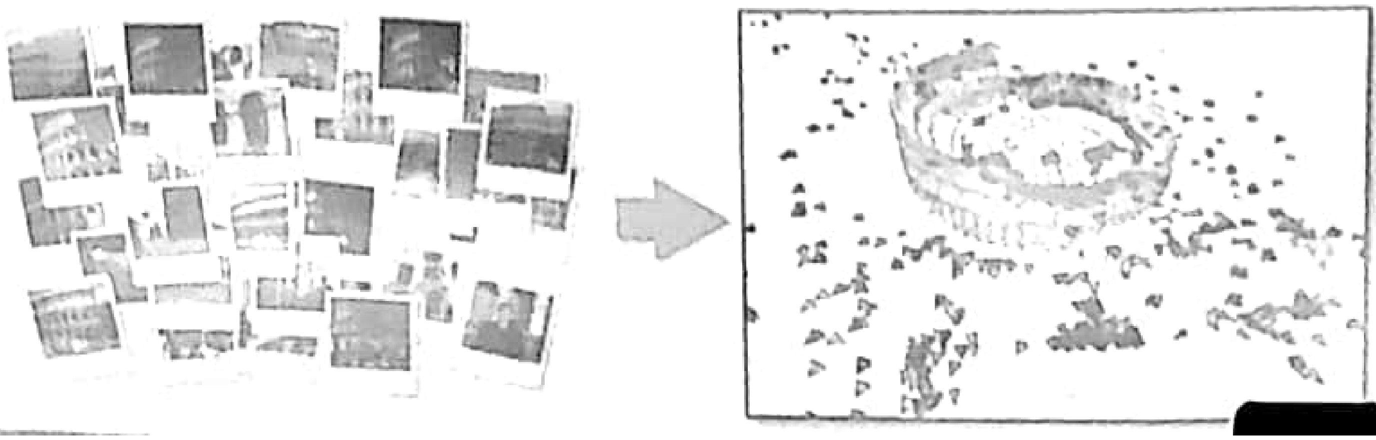
- Given $m$ views of $n$ 3D points, with unknown $\mathbf{P}_i$ and $\mathbf{X}_j$. Ideally, $\mathbf{x}_{ij} = \mathbf{P}_i\mathbf{X}_j$
- Minimize the re-projection error over all cameras/views:

$$\min_{P_i X_j} \sum_{i=1}^{m} \sum_{i=1}^{n} dist(X_{ij}, P_i X_j)^2$$

- A non-linear optimization problem. Can be solved using the Levenberg-Marquardt procedure directly.
- Called bundle adjustment. Known to photogrammetry community for a long time
- Needs good initialization as the complex non-linear optimization problem can get stuck in local minima

# Photo Tourism or Photo-Synth

- An automatic process, starting with independent images of a scene/monument/object. The images could be from a video sequence.
- Of particular interest has been SfM from Community Photo Collections (CPC), which are images that can be downloaded from flickr/picasa by giving a keyword like "Taj Mahal".



# SfM Steps

1. Download images for the place of interest!!
2. Extract descriptors from interest points on all images
3. Match points in pairs of images using Approximate Nearest Neighbours
4. Refine matches using Geometric Verification: Epipolar relationship, etc.
5. Form tracks of points across images. Transitively connect matches to get long matching "tracks"
6. Build image connectivity graph based on common points
7. Perform incremental SfM using the image connectivity graph and bundle adjustment

# Matching Points across Images

- Extract interest points $p_{ij}$ in each image $I_i$ and descriptors $s_{ij}$ for it. SIFT is popular. A few thousand in a typical image.
- Match interest points in image pairs. An approximate nearest neighbour approach is used, with a ratio test
- Point $p_{ij}$ matches point $p_{kl}$ iff:
  1. $dist(s_{ij}, s_{kl})$ is minimum over all points in $I_k$ and
  2. $dist(s_{ij}, s_{kl}) < r \times dist(s_{ij}, s_{km})$ where $p_{km}$ is the second closest point in $I_k$. $r$ is typically $0.6$
- Discard all points involved in case of multiple matches

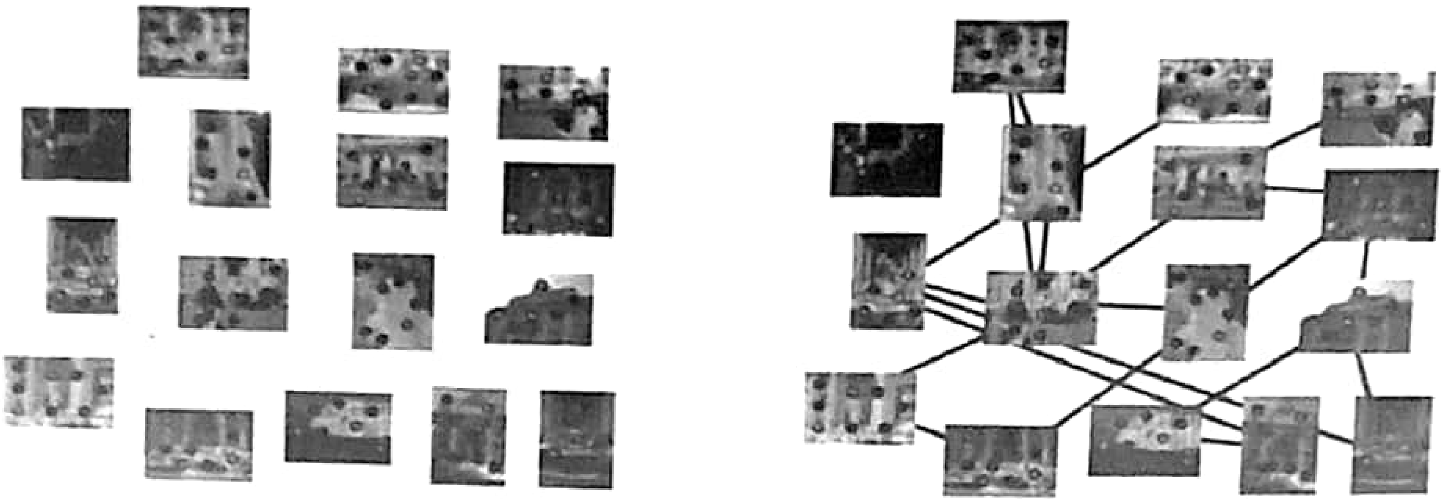
# Geometric Verification and Tracks

- Find fundamental matrix between pairs of cameras using RANSAC
- Refine matches by eliminating those not satisfying epipolar relation
- Propagate matches using transitivity to generate tracks, which represent the same world point in multiple images
- Form image connectivity graph. Two images have an edge if they share a point
- Densely connected regions represent parts of the scene visible to a large number of views. Sparse, leaf regions denote low sampling of parts of the scene

# Geometric Verification and Tracks

- Find fundamental matrix between pairs of cameras using RANSAC
- Refine matches by eliminating those not satisfying epipolar relation
- Propagate matches using transitivity to generate tracks, which represent the same world point in multiple images
- Form image connectivity graph. Two images have an edge if they share a point
- Densely connected regions represent parts of the scene visible to a large number of views. Sparse, leaf regions denote low sampling of parts of the scene

# Features and Graph Match



Form **tracks** of points by transitively connecting matches. They represent the same 3D point in multiple images.

# Track Statistics

- For a typical large data set with approx 3000 images:
  - 1.5 million tracks
  - 75-80% with of length 2
  - 98% of length less than 10
  - A few tracks of length more than 100
- Remember: Only 2D feature matching and verification has been done so far, but we seem to have come far!!

# Structure from Motion

- 11m + 3n parameters for m cameras and n points. 2mn equations mapping each point in each camera
- Recover camera and structure. Minimize reprojection error across all of them using a non-linear minimization step. This needs good initialization
- Not possible to do them all together. So, start with one pair of cameras and incrementally add more cameras
- Adjust points and cameras to reduce global reprojection error after new cameras are added

**Modern digital cameras store a lot of metadata in the images as EXIF tags, including the focal length!**

Assume: Only focal length is the unknown intrinsic parameter!

# Incremental SfM

- Find a strong starting pair of cameras. These should have a large number of points in common and a large baseline
- Find a pair with a large number of matches. Compute a planar homography from the matches. The pair is good if the error from the homography is high!
- Select the pair with the lowest percentage of inliers to homography using RANSAC
- Estimate the essential matrix for the camera pair
- Reconstruct cameras and common points using the essential matrix
- Perform bundle adjustment to minimize reprojection error

# Adding Views

- While there are more connected cameras
  - Pick an image that sees most number of 3D points so far
  - Estimate pose of the camera using DLT and known 3D points. Perform a local bundle adjustment to correct new camera pose
  - Triangulate new points (if any) and add to the collection
  - Perform a global bundle adjustment on all cameras and points, using a non-linear optimization step
- Can remove outlier tracks altogether
- Can add a small group of camera views together, instead of one at a time

# Bundle Adjustment

- Find P, X that minimizes (with visibility indicator $w_{ij}$)

$$g(P,X) = \sum_{i}^{m} \sum_{j}^{n} w_{ij} \left\| p_{ij} - P_i X_j \right\|^2$$

- Write it as $g(P,X) = \|A - \mathrm{P}(P,X)\|^2$, where P is the non-linear camera projection function
- Linearize
- Iterative solution using Levenberg-Marquardt method
- A sparse problem as the indicator $w_{ij}$ of point $j$ being visible in camera/image $i$ is sparse.

# Practical Aspects

- Heavy computations. Several days to reconstruct 500 images. About half of that time is for the bundle adjustment step
- Several optimizations have been worked on recently.
- Typical papers:
  - "Building Rome in a Day", ICCV 2009 (U of W)
  - "Building Rome on a Cloudless Day", ECCV 2010 (UNC)
- Combinatorics of pairwise matching is also huge. Use image search approaches to reduce the potential numbers

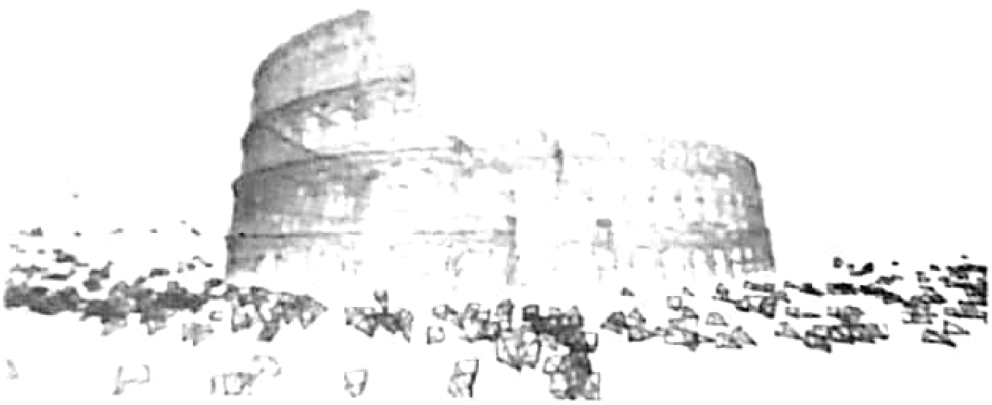
# Building Rome in a Day

Agarwal, Simon, Seitz, Szeliski. ICCV 2009

- Over a million images of the city of Rome
- Pair-wise matching can take 15 years at 2 pairs/sec
- Find 40 most similar words (fast matching)
- Query expansion to increase graph density
- Full bundle adjustment may run till end of time (nearly!)
- Use skeletal graphs to capture overall structure; perform bundle adjustment in local clusters
- Reconstructed Rome in 24 hours on a 1000-node cluster!!

- A local experiment on a 400-image Hampi dataset:
  - Extracting SIFT: 54 minutes,
  - Image matching: 17.2 hrs
  - Bundle adjustment: 12.6 hrs!!

# Rome Reconstructions



http://www.cs.cornell.edu/projects/bigsfm/

# Studios to Record Events

- Virtualized Reality (CMU, 1995)
  - A 51-camera recording setup
  - Off-line digitization
  - Multi-baseline stereo
  - Merge depth maps to get structure

# Studios to Record Events

- Virtualized Reality (CMU, 1995)
  - A 51-camera recording setup
  - Off-line digitization
  - Multi-baseline stereo
  - Merge depth maps to get structure
- Free-Viewpoint Video (ETH, MF
  - Multicamera setup, all digital
  - Visual hull for quick structure
- 123D from Autodesk
  - Submit your photographs
  - Get a 3D model!!