COMPUTER VISION AND PHOTOGRAMMETRY

SLAM

TEXT

SO FAR IN THIS CLASS

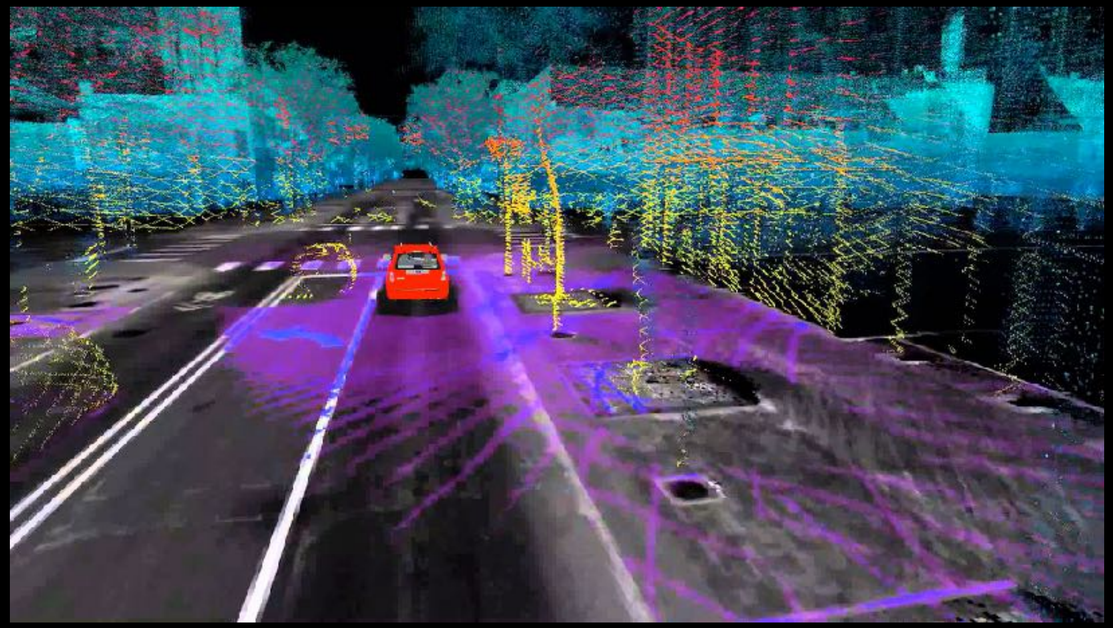
- Detect Keypoints
- Match them using Descriptors
- Recover Camera Motion
- Recover Structure
- Triangulate
- Texture

CONTENTS

SLAM OVERVIEW

WHAT IS SLAM

- Simultaneous Localization and Mapping
- A traditional problem in robotics

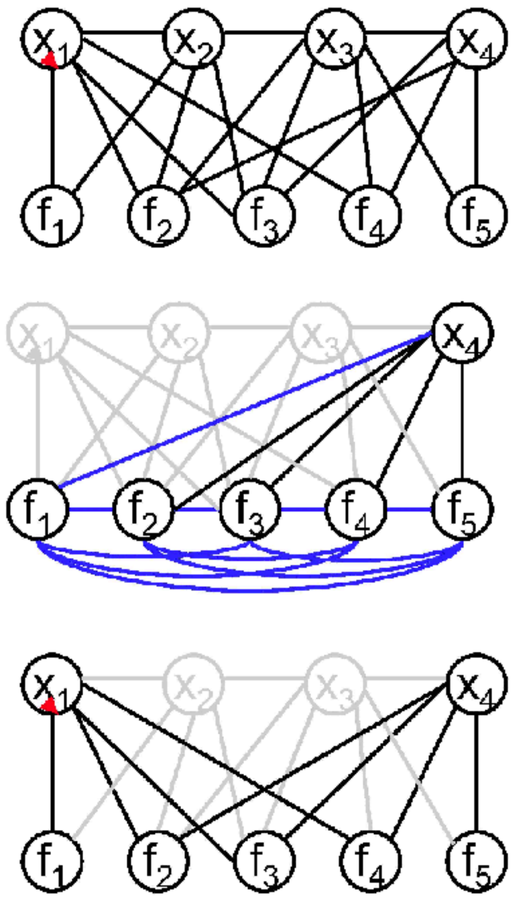


TYPES OF SLAM

- Landmark based
- LIDAR (1D, 2D, 3D)
- SONAR
- CAMERAS
  - Also RGB-D cameras
  - WIFI, radar,

VISUAL SLAM

- Uses cameras
- Similar problem to Structure from Motion
- Different Constraints
  - Real Time perfomance
  - From Video - Small baseline - Lower Resolution
  - Doesn’t care that much about geometry



- Original SLAM problem
- EKF approach
  - Only keeps the last pose
  - $O(n^2)$ with the number of features
  - Limited to 200-300 features in real-time
- Keyframe approach (PTAM)
  - Uses only a few keyframes for map estimation with non-linear optimization
  - Can handle thousands of points
  - Given the same computational effort is more precise than EKF-SLAM

ORB-SLAM

ORB-SLAM: A VERSATILE AND ACCURATE MONOCULAR SLAM SYSTEM.

Some slides and images taken from the authors

FEATURE BASED VISUAL SLAM

- Initialization
- Feature detection
- Feature Matching
- Camera position estimation
- Relocation and Loops

INITIALISATION

- Need a starting point
  - Track a known object
  - Initialize by moving the camera, select two good frames
    - Need to detect F or H (planar surface)
  - If RGB-D Cameras, use the first depth map

ORB FEATURES

WHAT FEATURE DETECTOR?

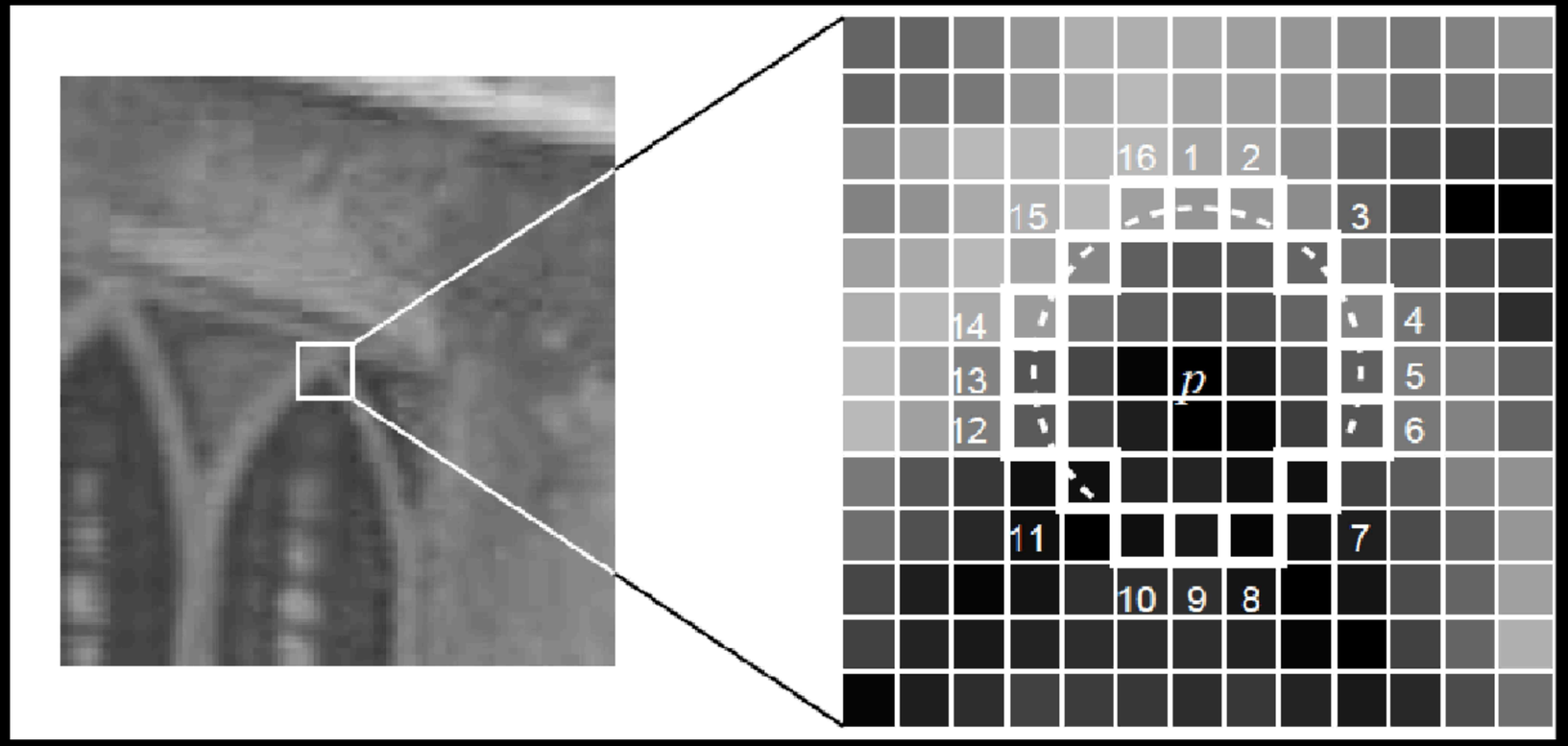
- Repeatability
- Accuracy
- Invariance
  - Illumination (auto-adjust gain change in cameras)
  - Position and rotation
  - Scale and viewpoint
- EFFICIENCY

FEATURES

- Fast corners
- At multiple scales (8)
- To ensure good distribution - grid and try to get 5 features in each
- 1000 - 2000 features

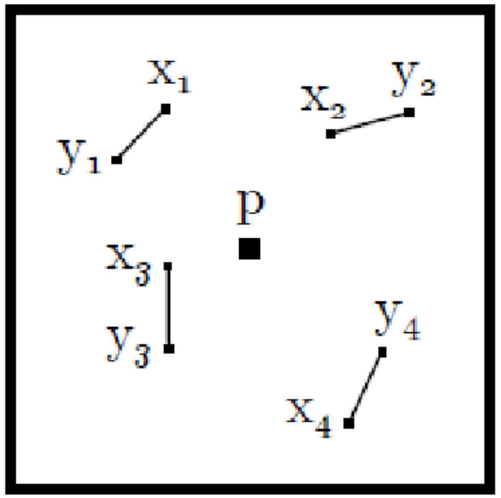
FAST CORNERS

- Pixel p surrounded by n consecutive pixels all brighter (or darker) than p
- Much faster than other detectors



Rosten and Drummond; https://www.edwardrosten.com/work/rosten_2006_machine.pdf

BRIEF



$$D_i(\mathbf{p}) = \begin{cases} 1 & \text{if } I(\mathbf{p}+\mathbf{x}_i) < I(\mathbf{p}+\mathbf{y}_i) \\ 0 & \text{otherwise} \end{cases}$$

$$\hookrightarrow D(\mathbf{p}) = [\,1\;0\;0\;1\;1\;0\;0\;0\;1\ldots\,]$$

- 256 binary string
- Not invariant to rotation

ORB

- FAST + BRIEF with orientation compensation
- Alternative to SIFT
- Moment of a Patch

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y),$$

- Centroid

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$$

- Vector |OC| where O is the center of the patch and C its centroid
- Orientation

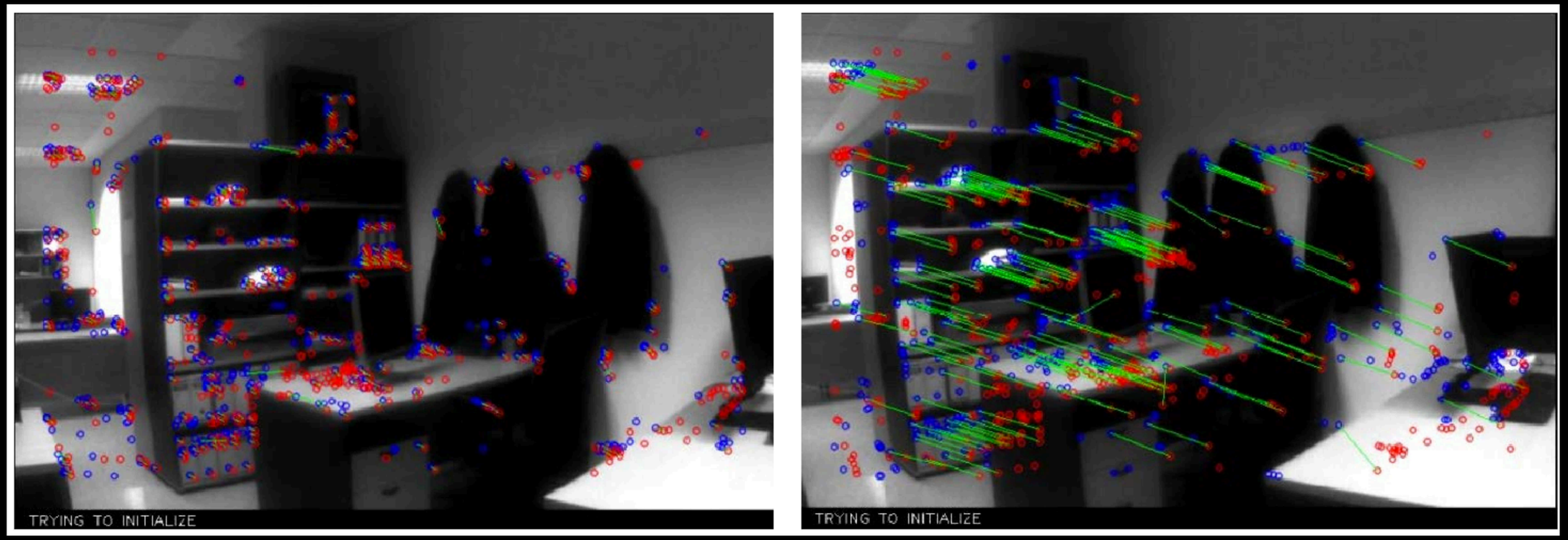
$$\theta = \text{atan2}(m_{01}, m_{10}),$$

TEXT

Detector	Descriptor	Rotation Invariant	Automatic Scale	Accuracy	Relocation & Loops	Efficiency
Harris	Patch	No	No	++++	-	++++
Shi-Tomasi	Patch	No	No	++++	-	++++
SIFT	SIFT	Yes	Yes	++	++++	+
SURF	SURF	Yes	Yes	++	++++	++
FAST	BRIEF	No	No	+++	+++	++++
ORB	ORB	Yes	No	+++	+++	++++

E OR H

- Camera is pre calibrated
- Use Essential Matrix (5 -8 point) or Homography (4 points)



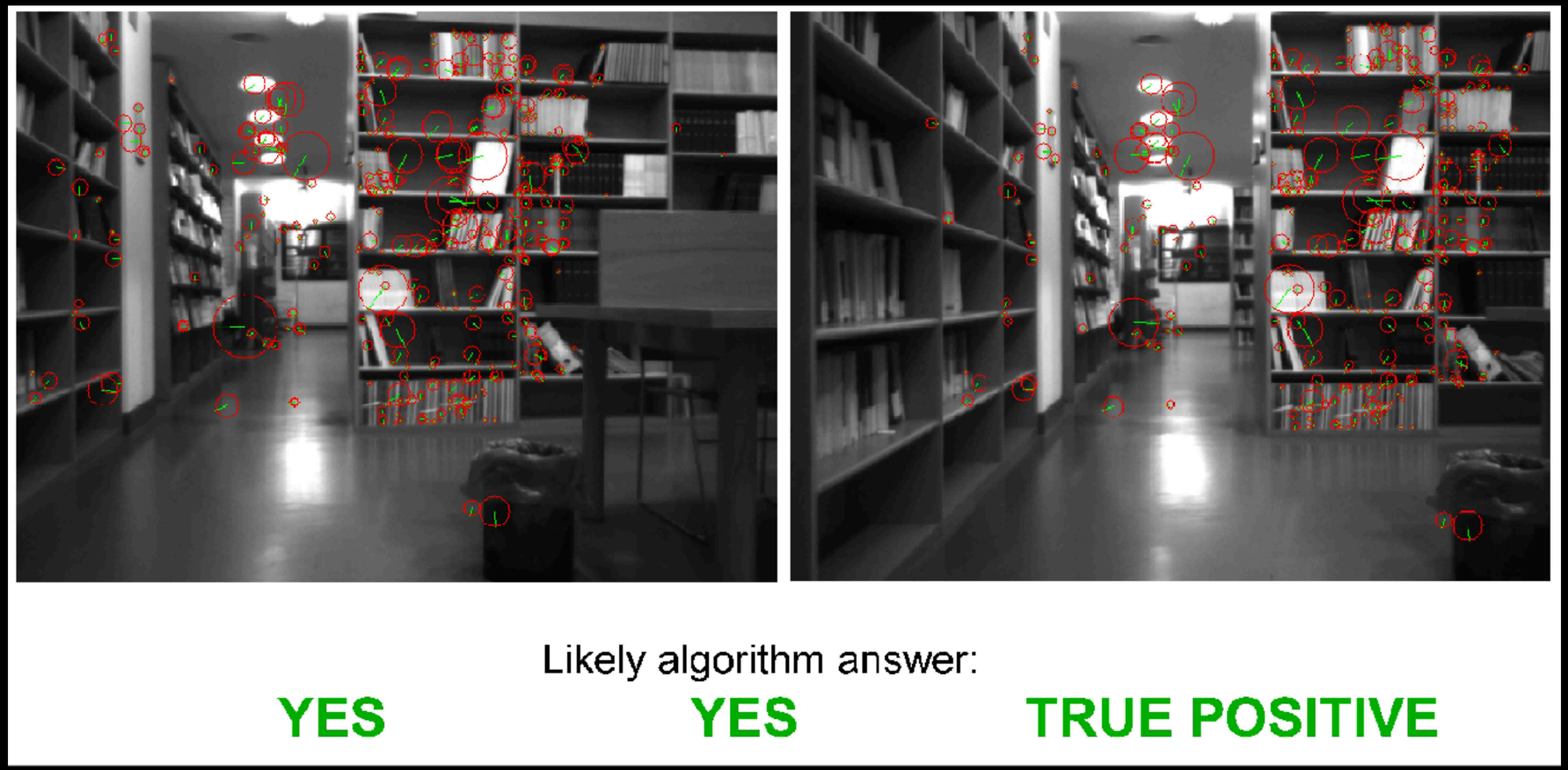
RELOCATION PROBLEM

RELOCATION AND LOOP CLOSING

- Relocation problem
  - SLAM Tracking lost: occlusion, no features, too quick motions
  - Reaquire the position and continue
- Loop Closing
  - You go back to a mapped area
  - Loop detection: Avoid duplicate mapping
  - Loop correction: Correct accumulated error (drift)
- BOTH NEED A PLACE RECOGNITION TECHNIQUE

LOOP DETECTION

- IS THIS A LOOP?



LOOP DETECTION

- IS THIS A LOOP?

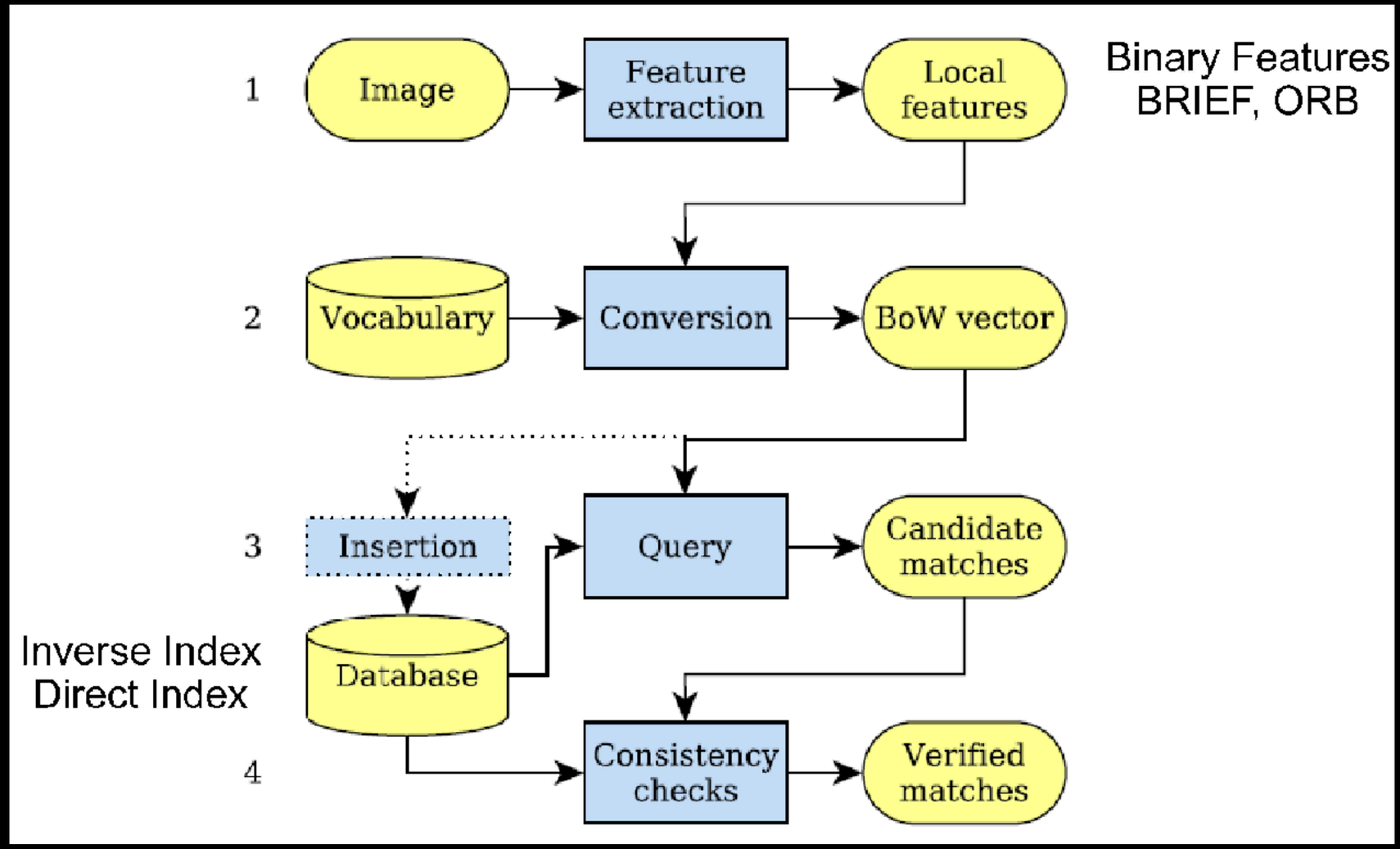
Likely algorithm answer:

NO YES FALSE POSITIVE

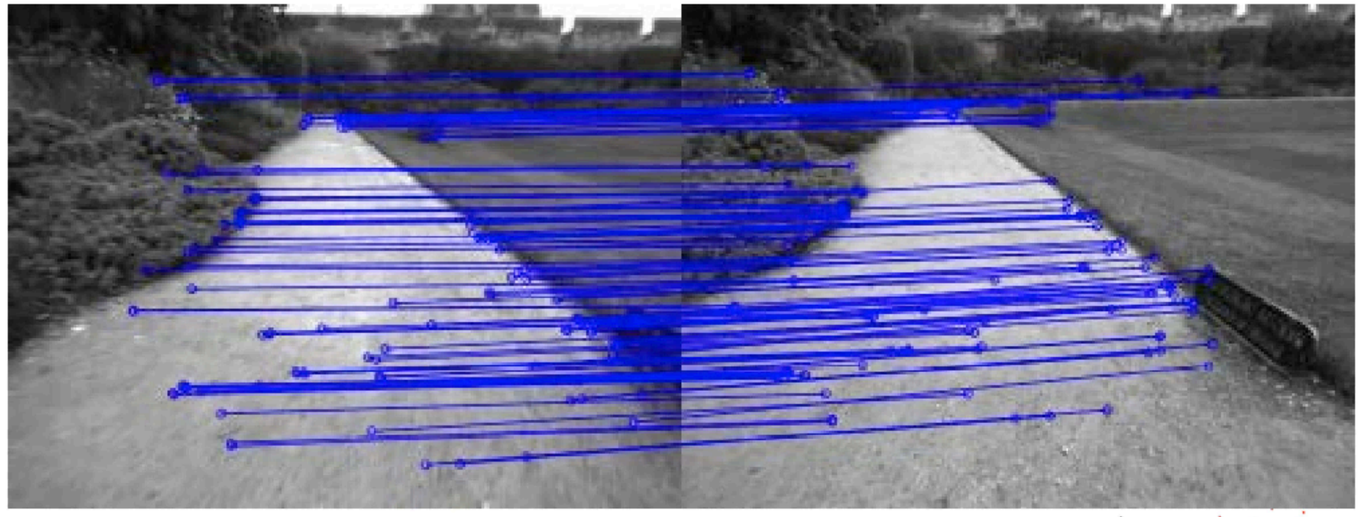
DETECTING PLACES

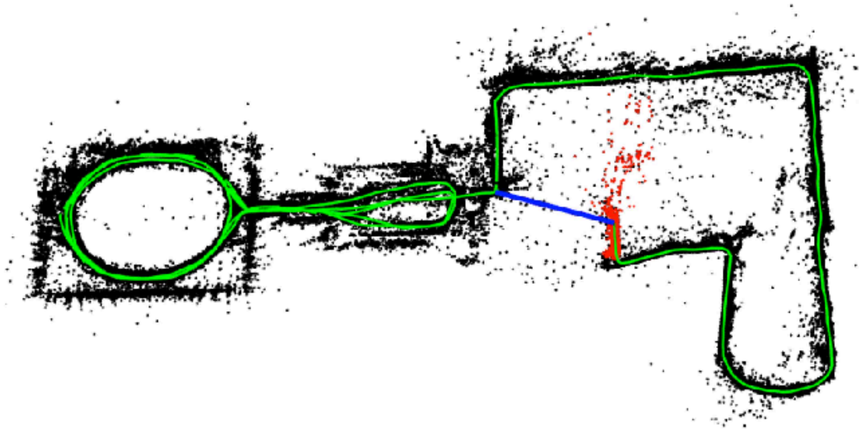
- Keep a database of features
- The bag-of-words model is a simplifying representation used in natural language processing and information retrieval (IR). In this model, a text (such as a sentence or a document) is represented as the bag (multiset) of its words, disregarding grammar and even word order but keeping multiplicity. (WIKIPEDIA)

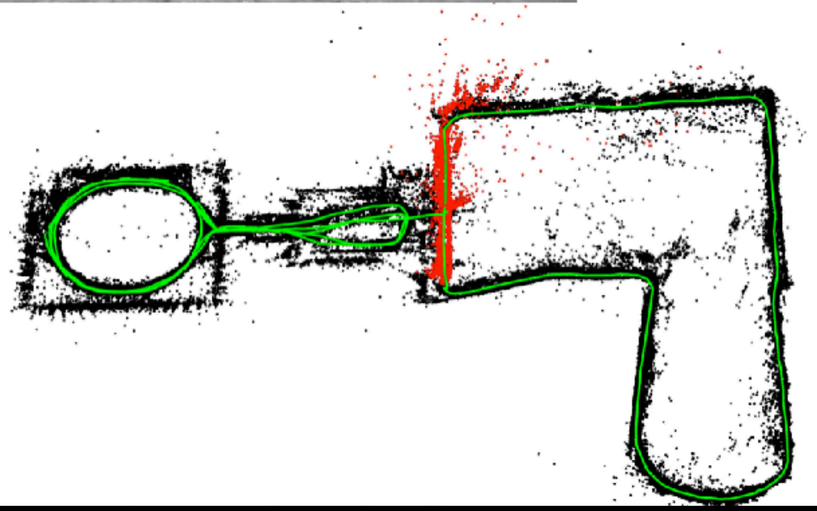
BAG OF WORDS APPROACH



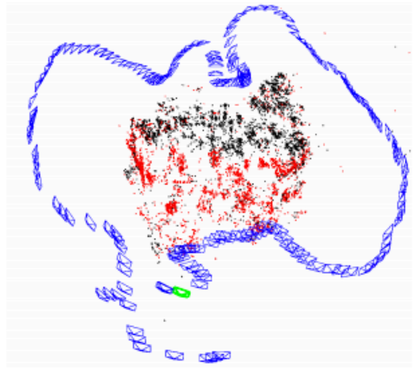
LOOP CORRECTION



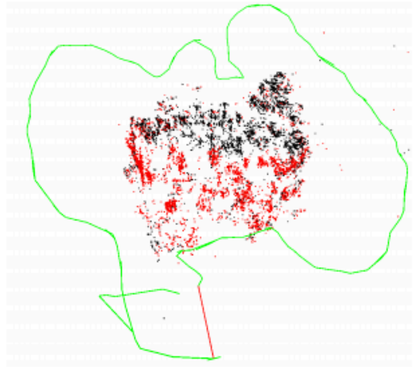




COVISIBILITY GRAPH

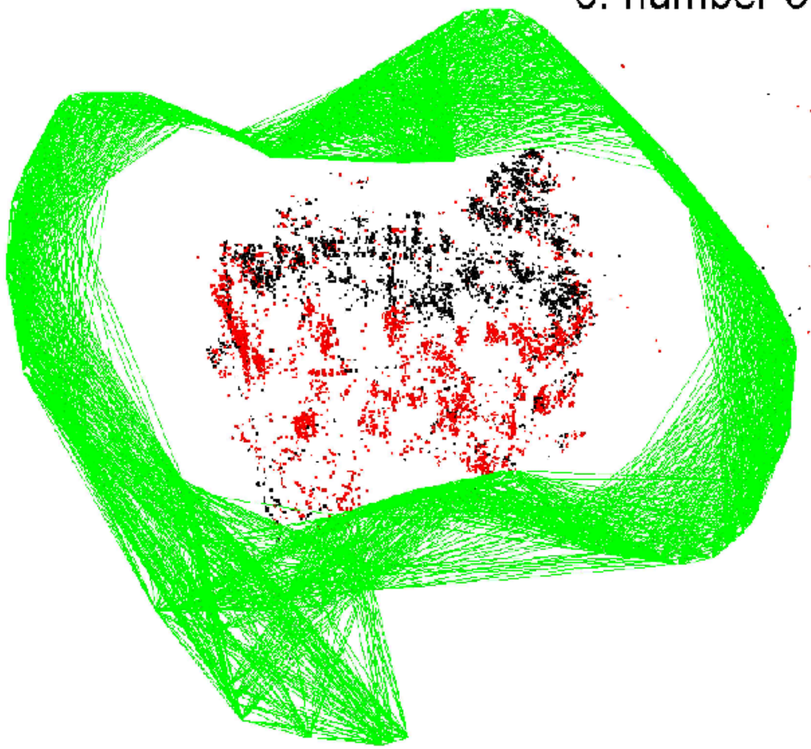


(a) KeyFrames (blue), Current Camera (green), MapPoints (black, red), Current Local MapPoints (red)



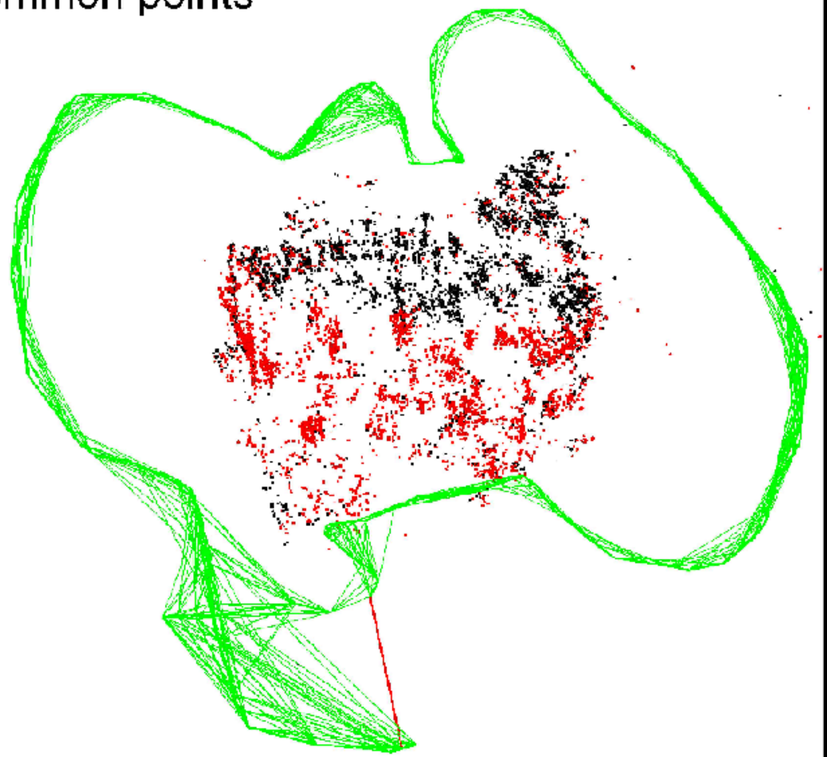
(c) Spanning Tree (green) and Loop Closure (red)

θ: number of common points



$\theta_{min} = 15$

Used for Local BA



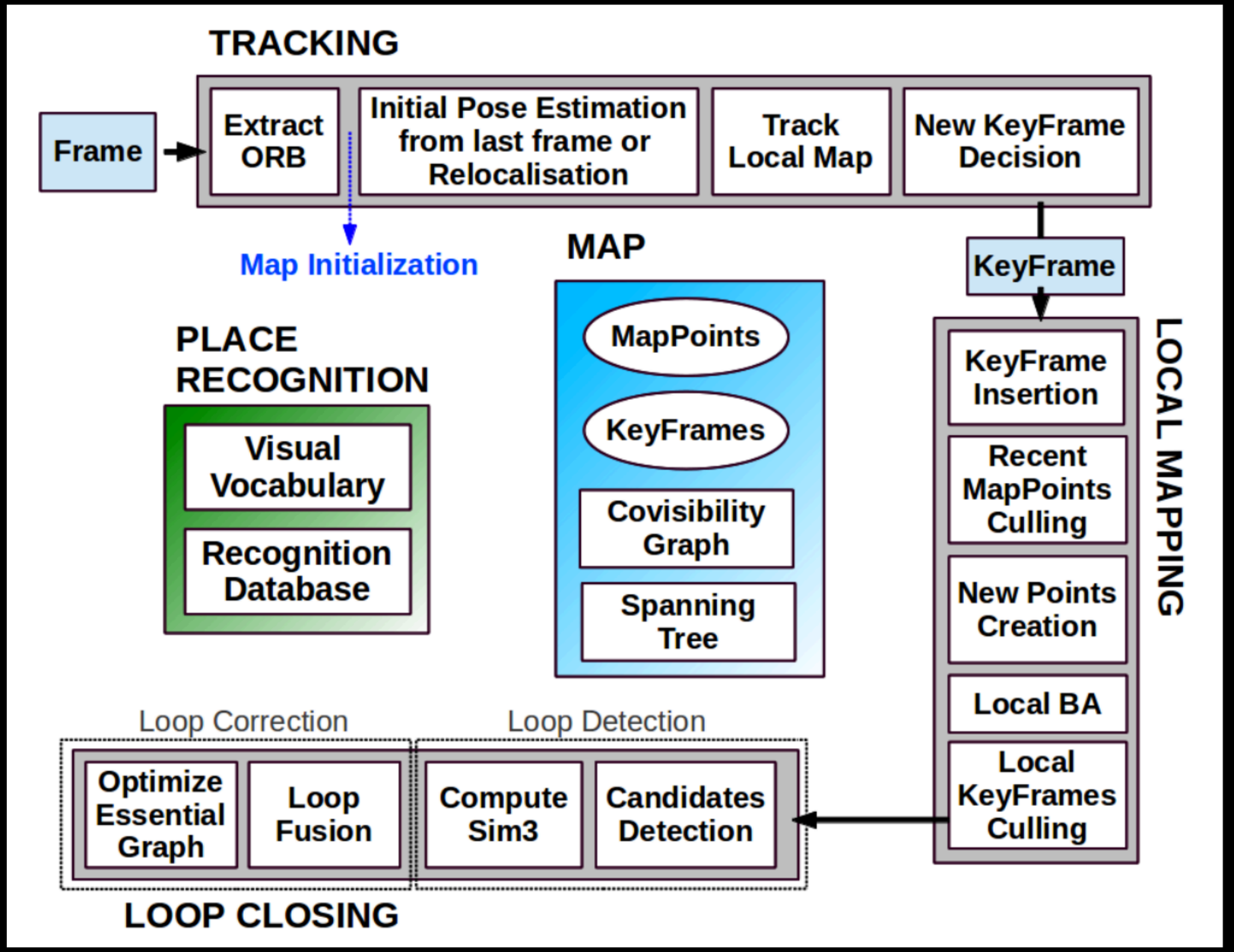
$\theta_{min} = 100$

Used for Loop Correction

ALSO BUNDLE ADJUSTMENT

- Local BA
- Only-Motion BA
- Full BA in parallel

OVERALL

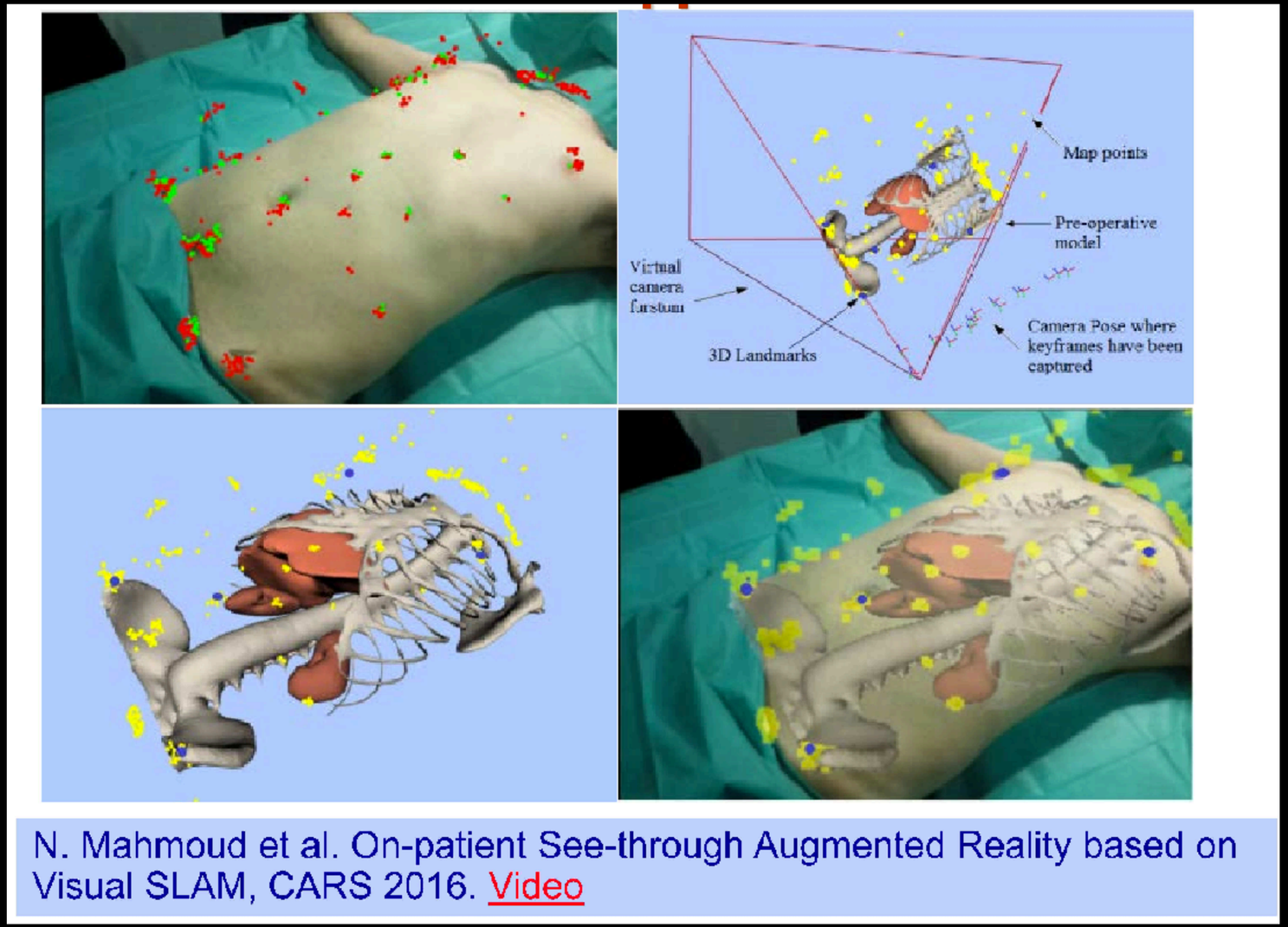

TRACKING
Frame
Extract ORB
Initial Pose Estimation from last frame or Relocalisation
Track Local Map
New KeyFrame Decision
Map Initialization
MAP
MapPoints
KeyFrames
Covisibility Graph
Spanning Tree
PLACE RECOGNITION
Visual Vocabulary
Recognition Database
KeyFrame
LOCAL MAPPING
KeyFrame Insertion
Recent MapPoints Culling
New Points Creation
Local BA
Local KeyFrames Culling
Loop Correction
Loop Detection
Optimize Essential Graph
Loop Fusion
Compute Sim3
Candidates Detection
LOOP CLOSING

APPLICATIONS

AUGMENTED REALITY: GAMES

- lets play pokemon

AUGMENTED REALITY: MEDICAL IMAGING



N. Mahmoud et al. On-patient See-through Augmented Reality based on Visual SLAM, CARS 2016. Video

ROBOTICS AND NAVIGATION

AR HOLOLENS-SIMILAR

SOME SYSTEMS

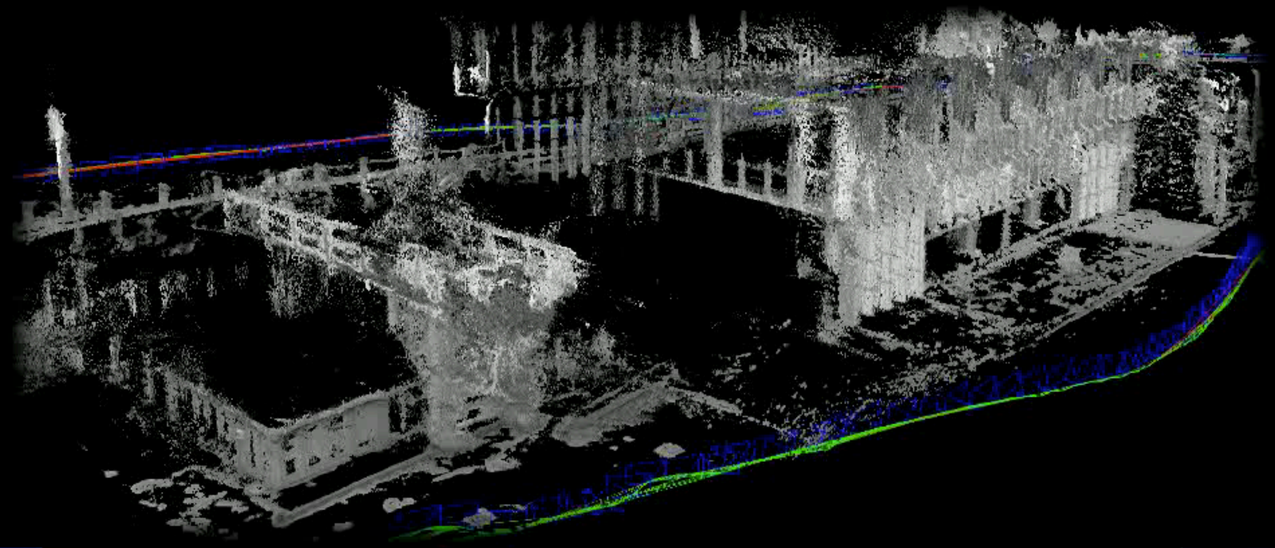
SOME SYSTEMS:ORB-SLAM2

SOME SYSTEMS:LSD-SLAM DIRECT

LSD-SLAM: Large-Scale Direct Monocular SLAM

Jakob Engel, Thomas Schöps, Daniel Cremers

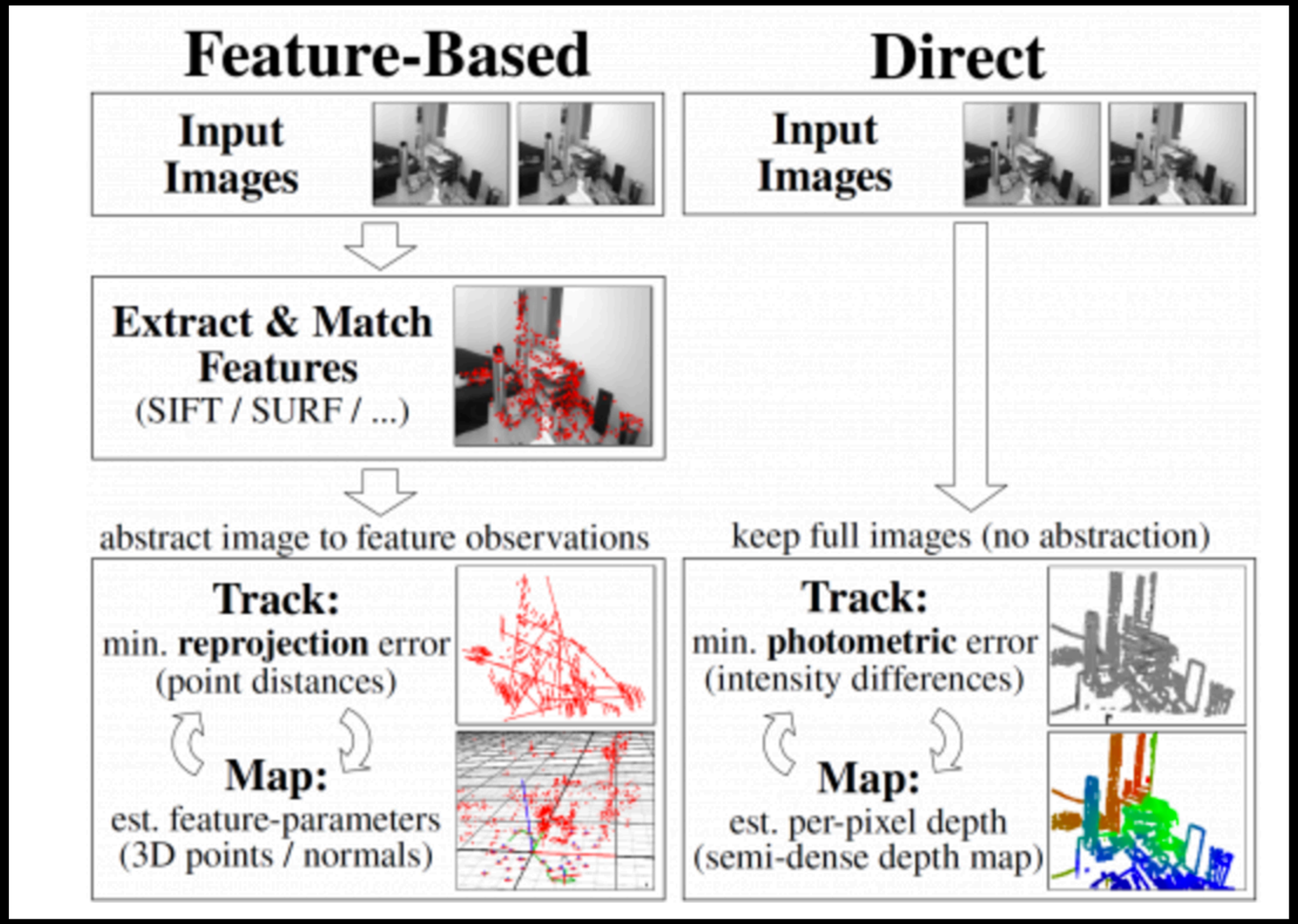
ECCV 2014, Zurich



Computer Vision Group
Department of Computer Science
Technical University of Munich

TEXT

DIRECT METHODS



http://www.dis.uniroma1.it/~labrococo/tutorial_icra_2016/icra16_slam_tutorial_engel.pdf

OPENSLAM.ORG

- https://openslam.org/
- https://github.com/raulmur/ORB_SLAM2
- https://github.com/Oxford-PTAM/PTAM-GPL

KINECT FUSION

KINECT FUSION

TEXT

KINECT FUSION

SIGGRAPH Talks 2011

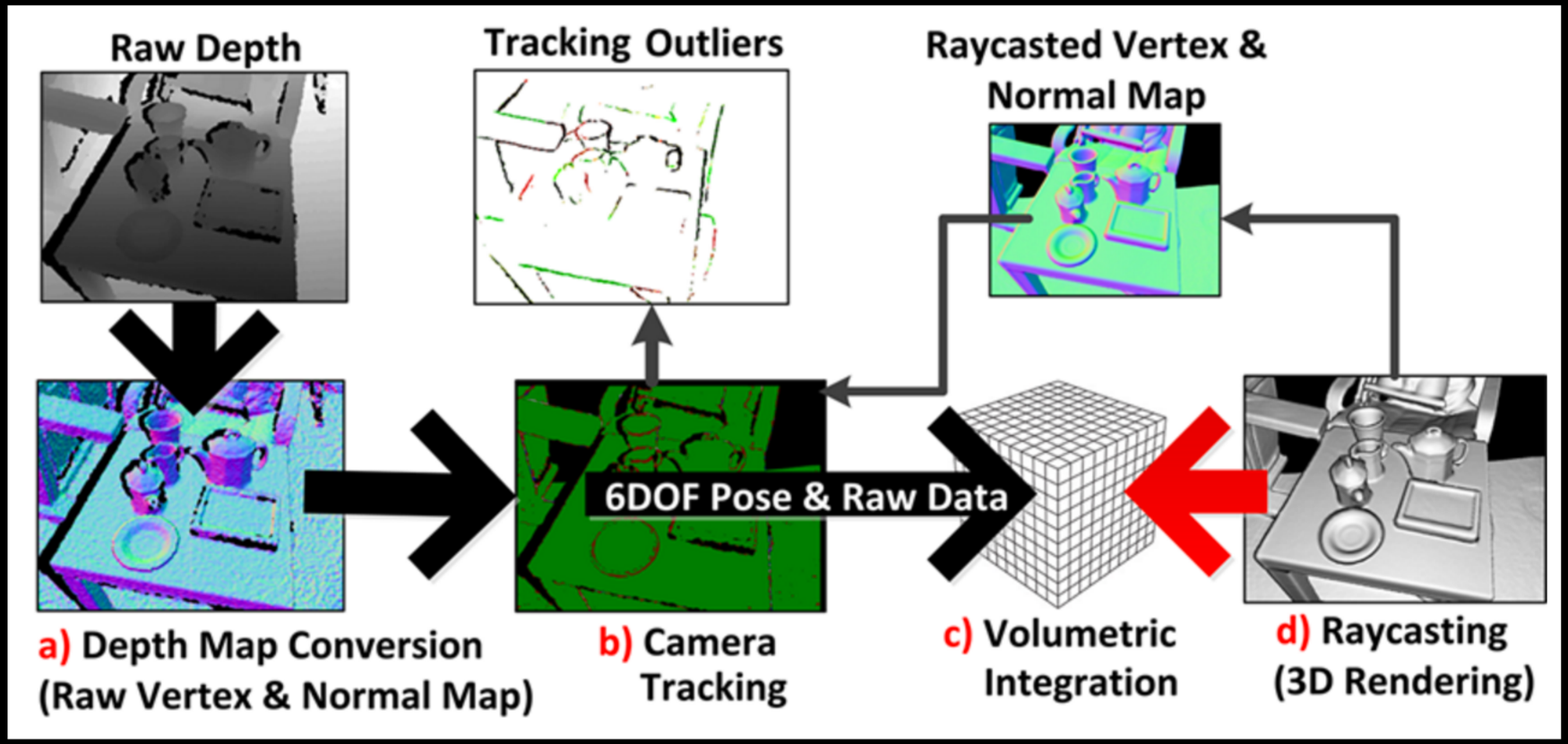
KinectFusion:

Real-Time Dynamic 3D Surface Reconstruction and Interaction

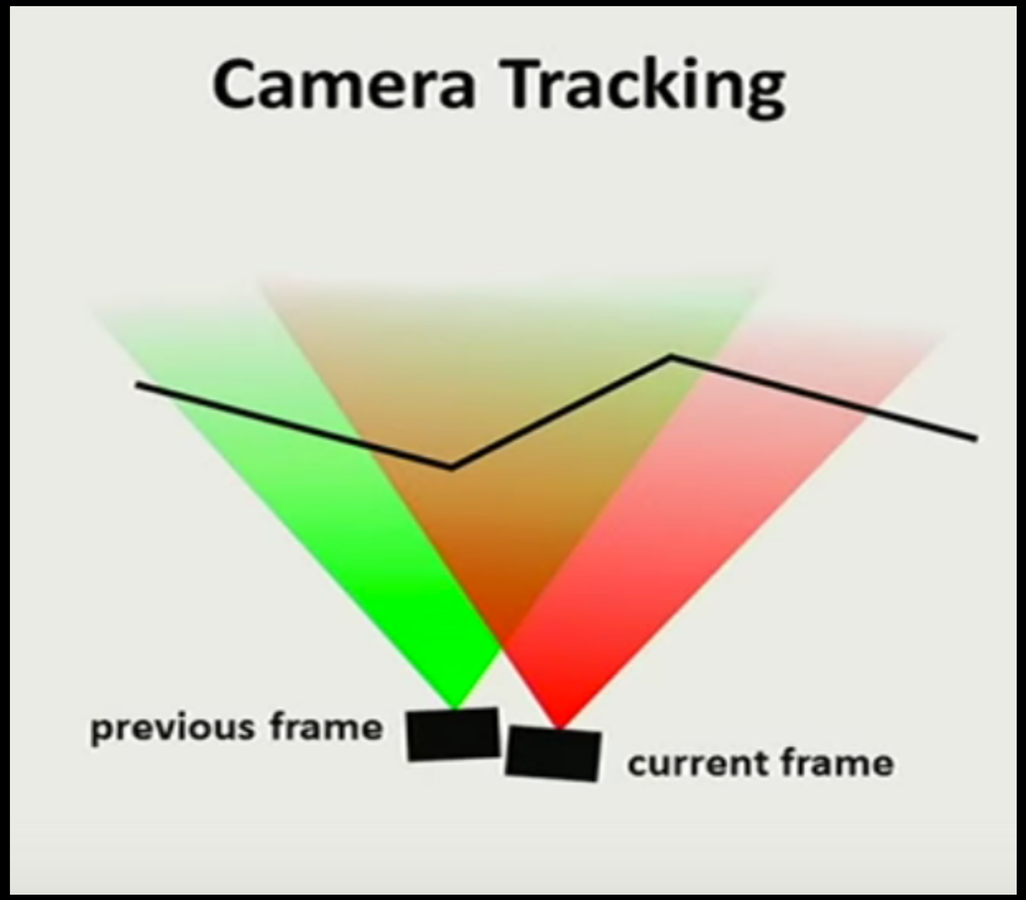
Shahram Izadi 1, Richard Newcombe 2, David Kim 1,3, Otmar Hilliges 1,
David Molyneaux 1,4, Pushmeet Kohli 1, Jamie Shotton 1,
Steve Hodges 1, Dustin Freeman 5, Andrew Davison 2, Andrew Fitzgibbon 1

1 Microsoft Research Cambridge 2 Imperial College London
3 Newcastle University 4 Lancaster University
5 University of Toronto

KINECT FUSION



TRACKING CAMERA



TRACKING CAMERA

- Iterative Closest Point (ICP)
  - NOT INSANE CLOWN POSSE
  - Typically used for aligning pointcloud meshes
  - Assumption: Rough aligment
    - Thats fine in this case

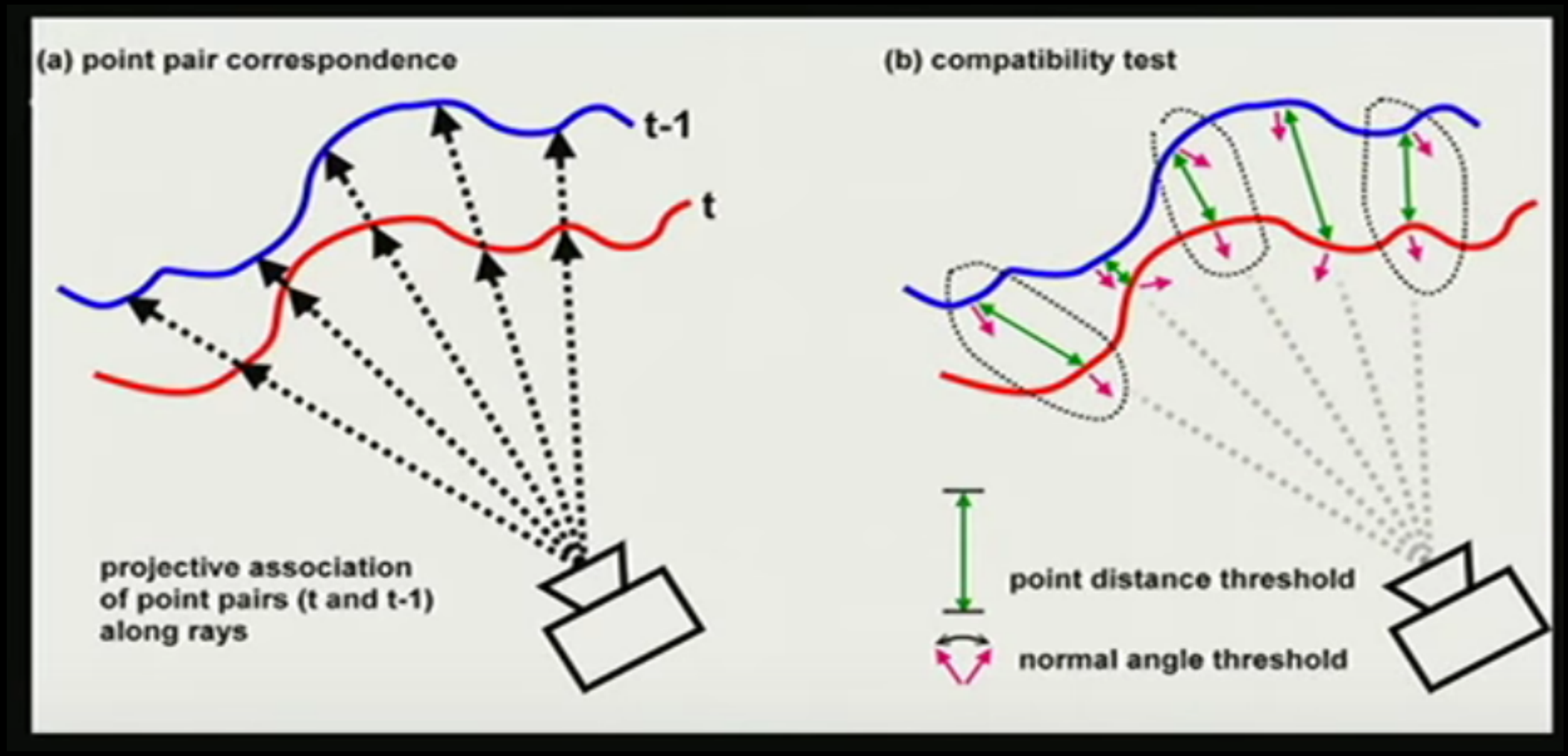
ICP FOR CAMERA TRACKING

- Points from same image positions
- Check for angles and distances: reject outliers
- Find a transformation that minimize energy function (sum of distances between points)
- Apply Transformation
- Iterate (about 5 times per frame, takes about 3ms in the GPU)

ICP FOR KINECT

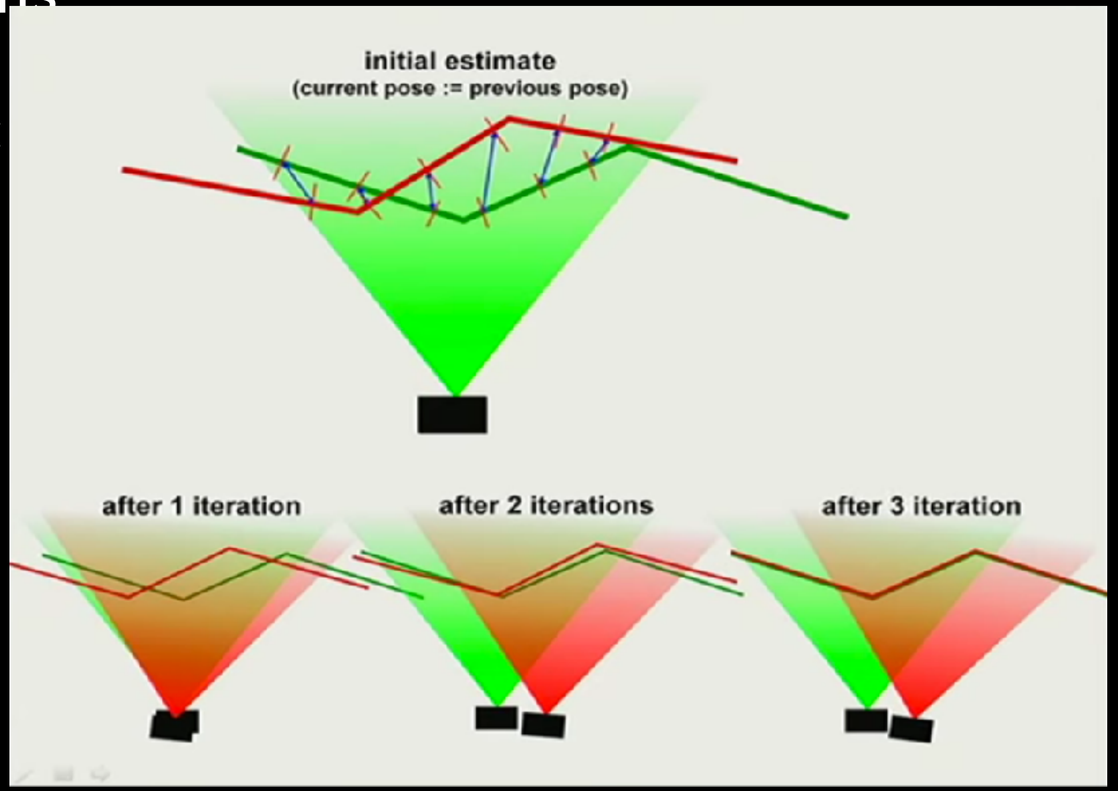
- it’s structured (stored in a grid)
- the intrinsic parameters of the camera that produced the frame are known
- the difference in position and rotation between subsequent frames is small
- points normals may be computed for the registered frames
- Point-to-Plane distance

ICP FOR CAMERA TRACKING

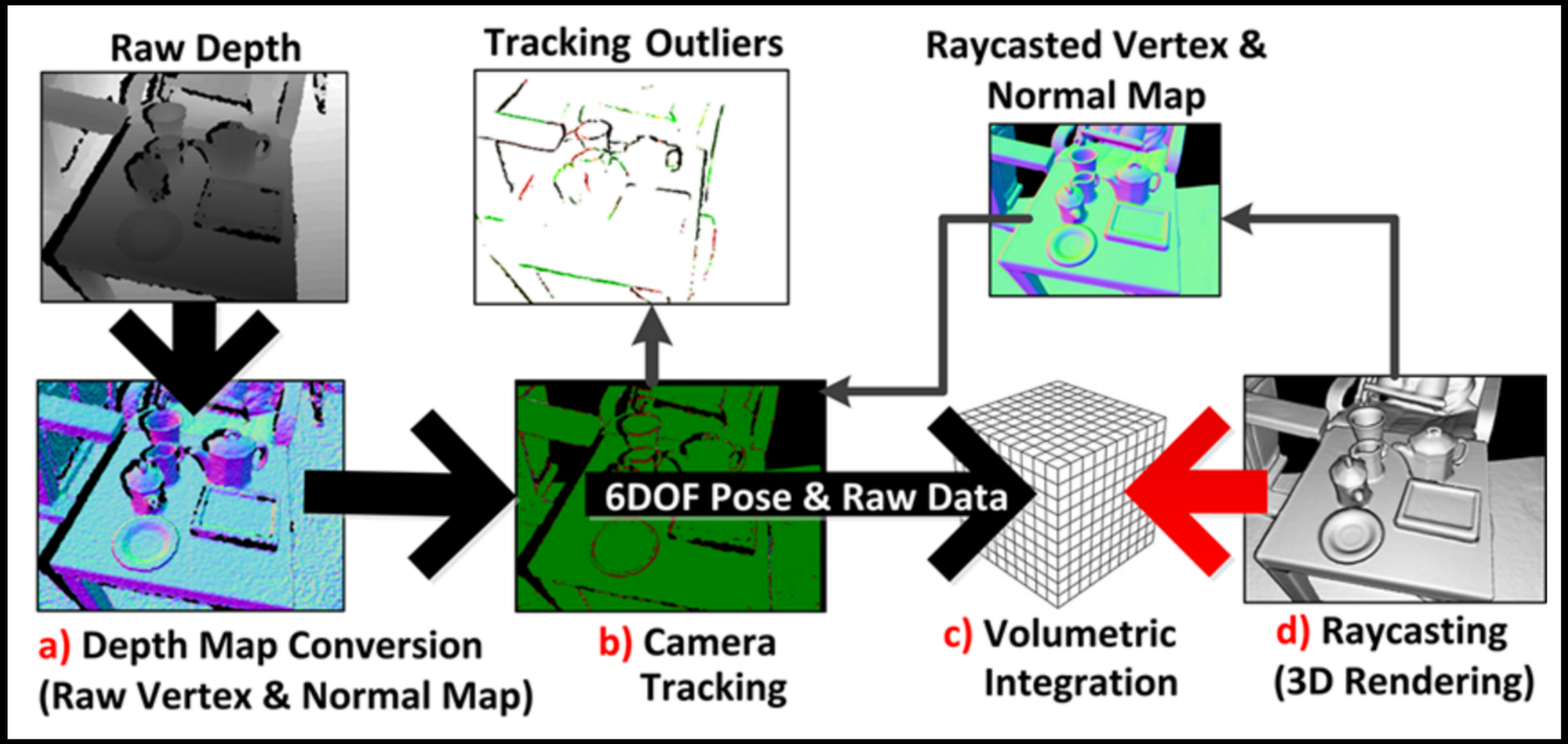


ICP FOR CAMERA TRACKING

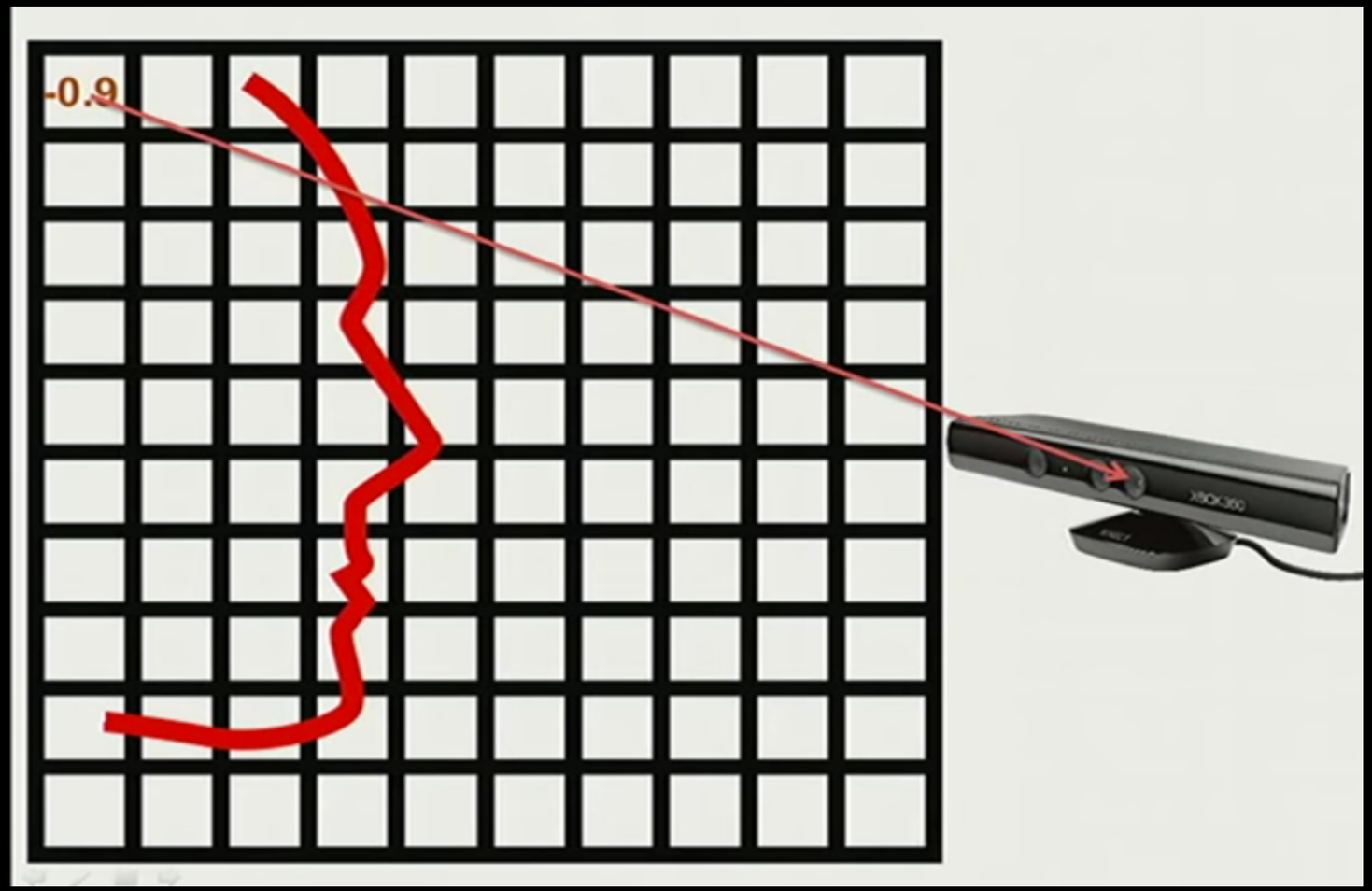
- Points from same image positions
- Check for angles and distances:



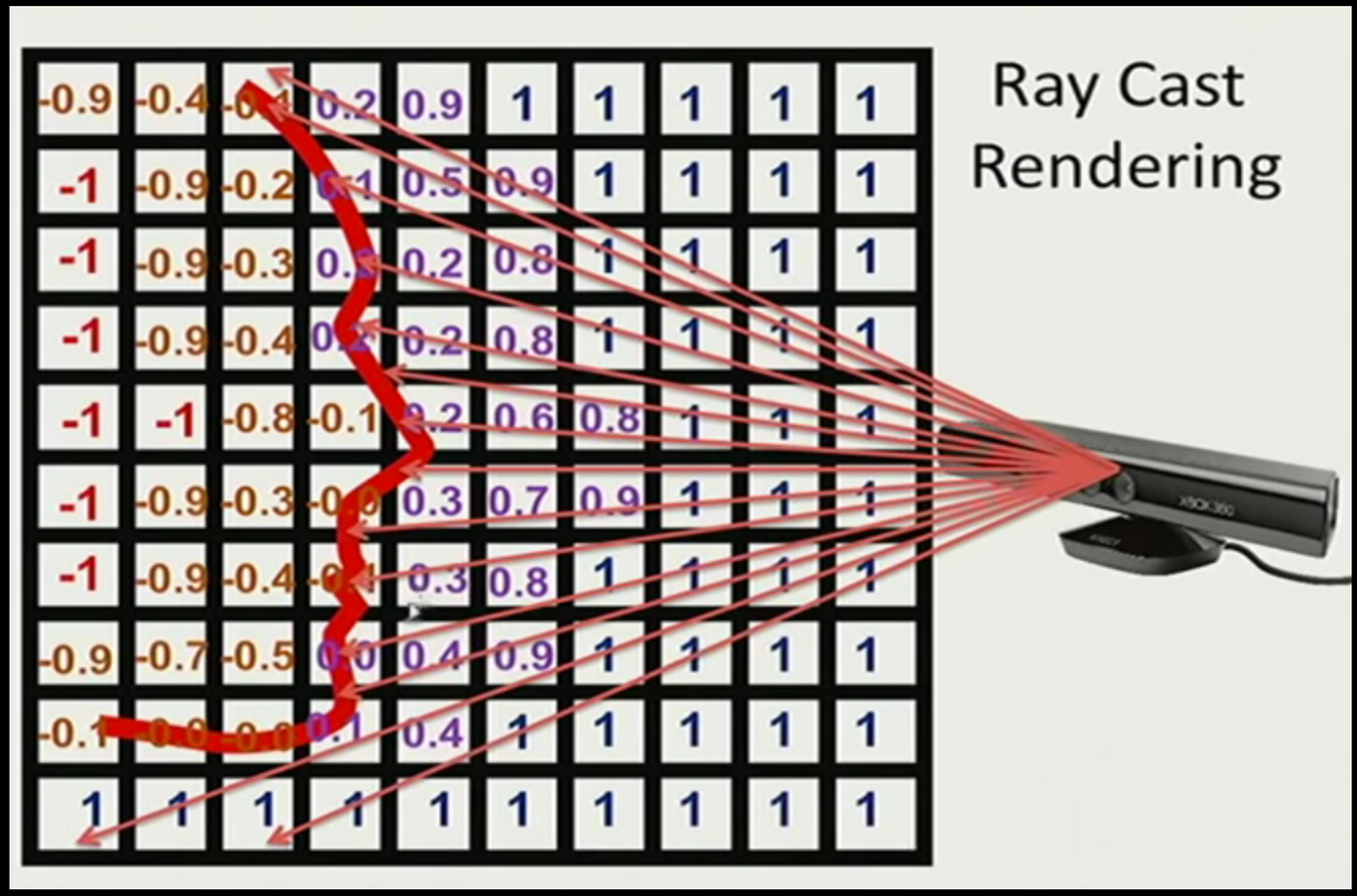
KINECT FUSION



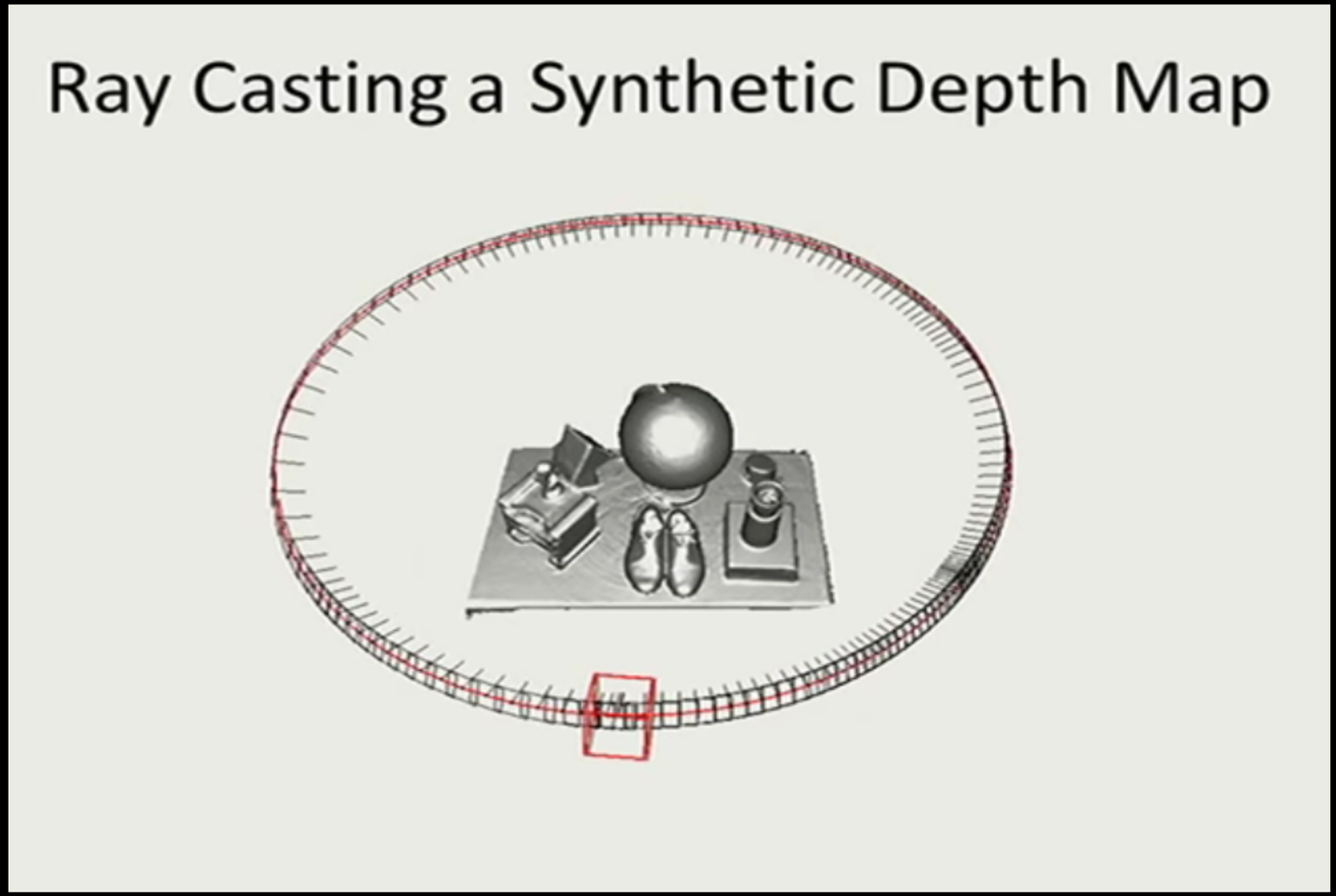
KINECT FUSION–DATA FUSION



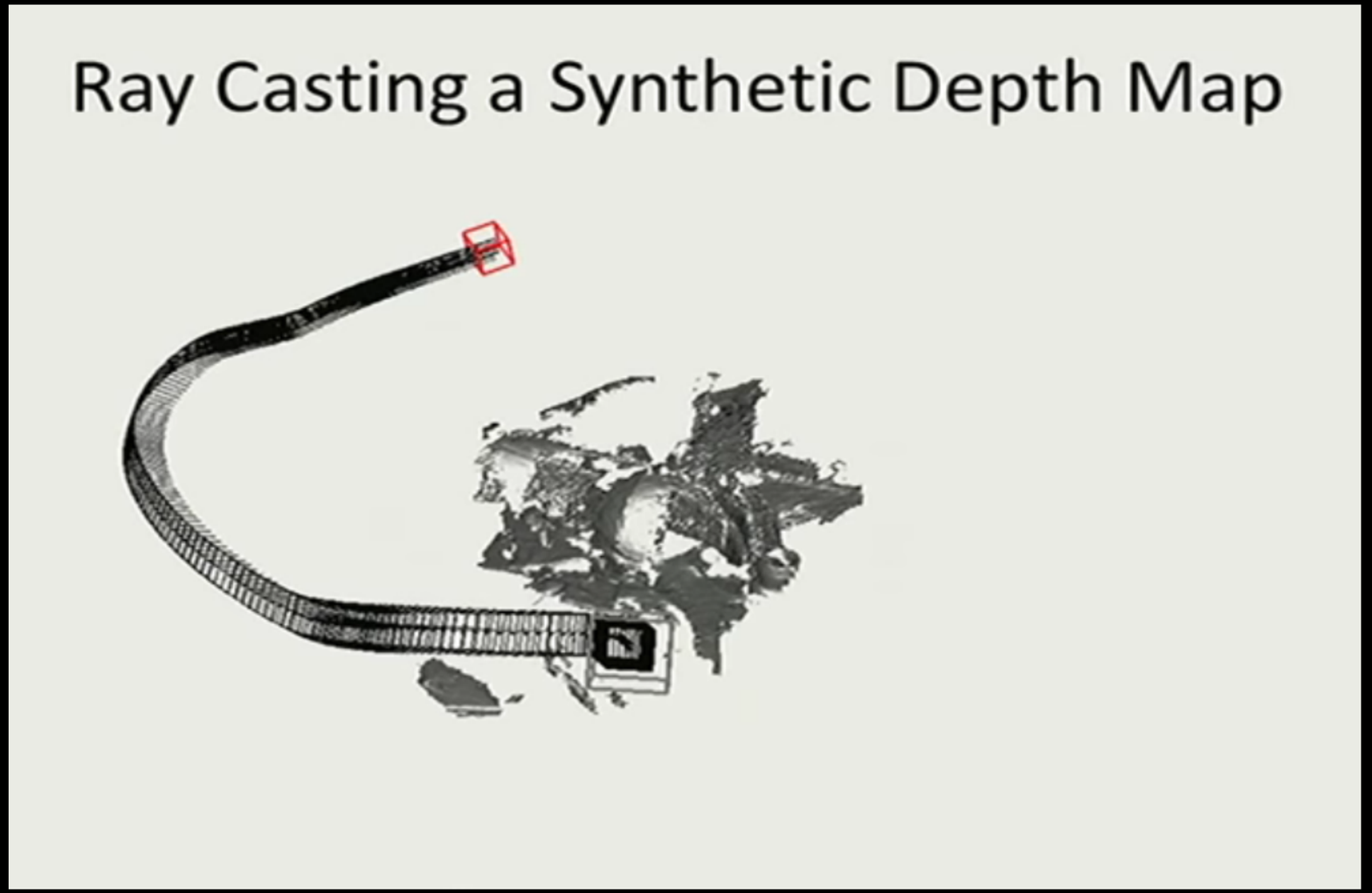
KINECT FUSION–DATA FUSION



POISSON SURFACE RECONSTRUCTION



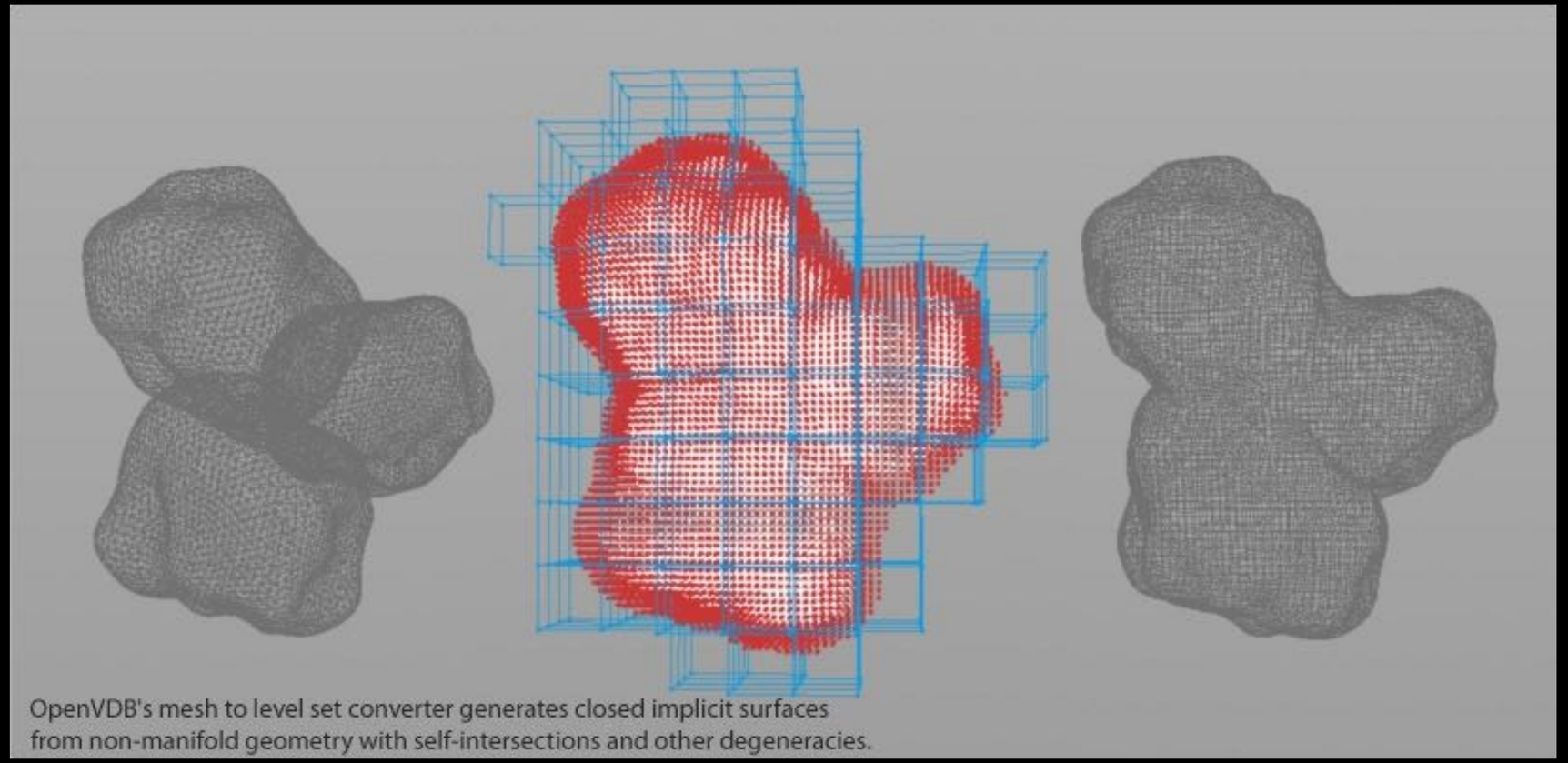
KINECT FUSION–DATA FUSION



KINECT FUSION–DATA FUSION

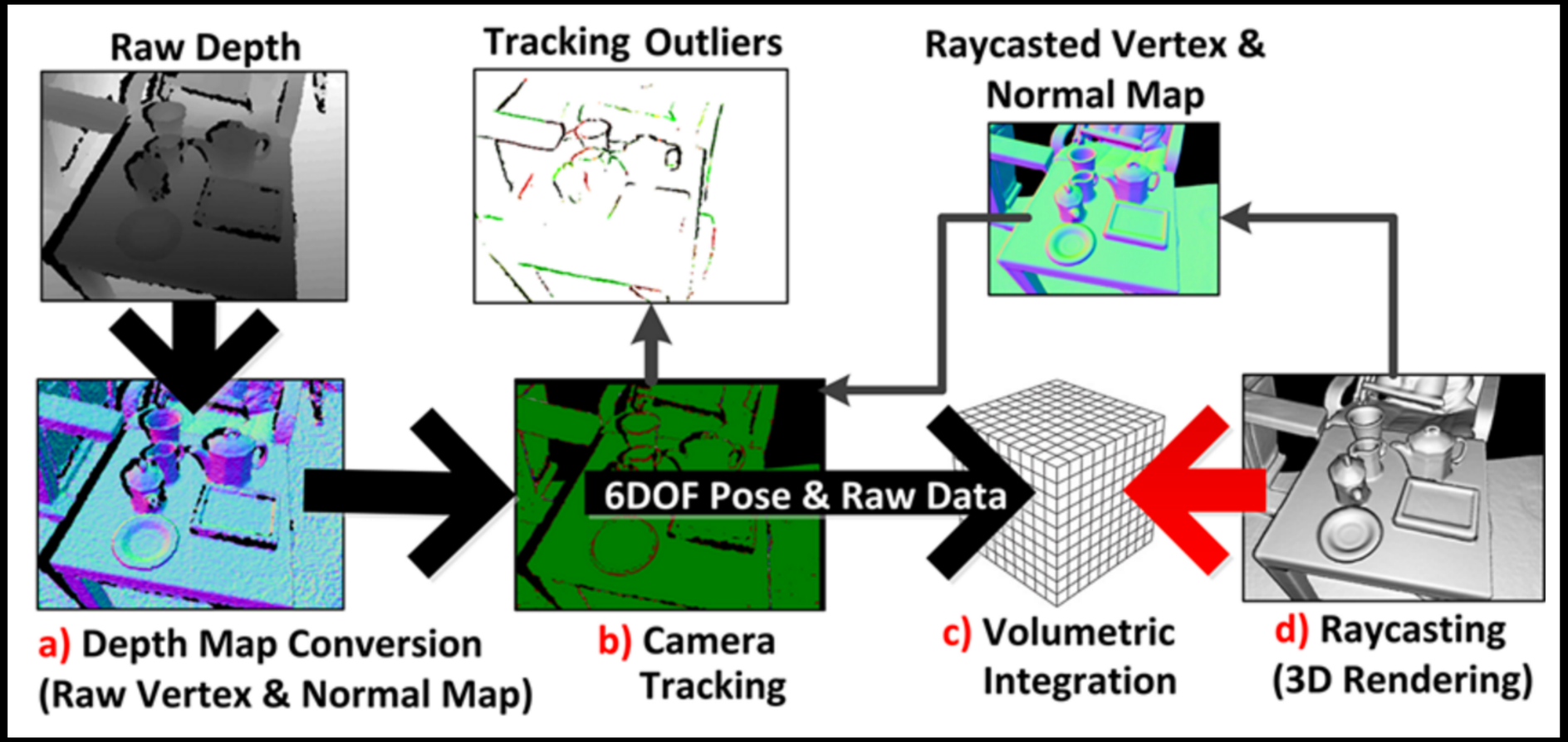
- Implicit Surfaces modeled with Truncated Signed Distance Function (TSDF)
- Curless & Levoy 1996
- Can be fusion easily. (simple average per voxel)

TEXT



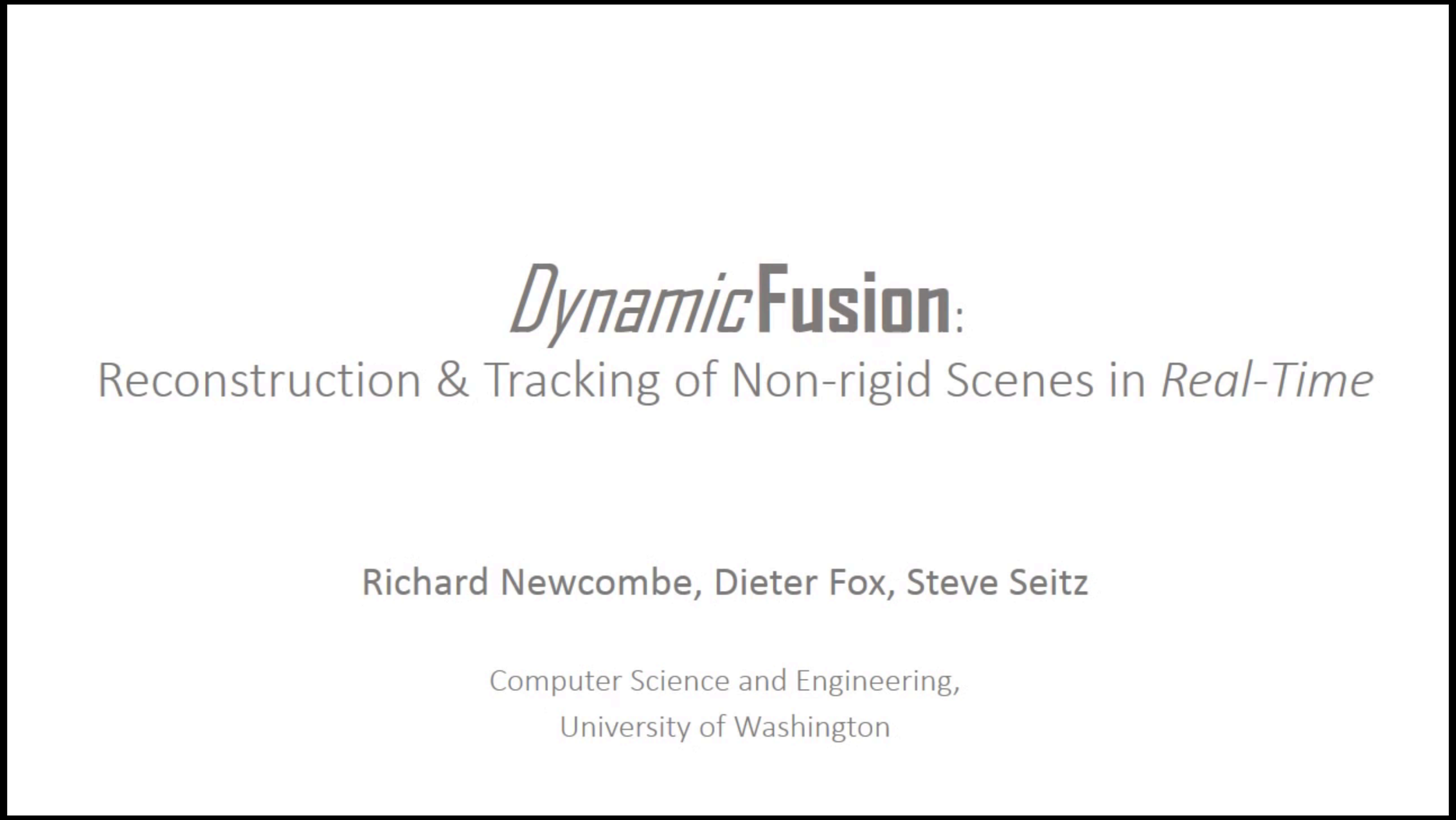
OpenVDB's mesh to level set converter generates closed implicit surfaces
from non-manifold geometry with self-intersections and other degeneracies.

KINECT FUSION



TEXT

DYNAMIC FUSION



DYNAMIC FUSION

(a) Initial Frame at $t = 0s$

(b) Raw (noisy) depth maps for frames at $t = 1s, 10s, 15s, 20s$

(c) Node Distance

(d) Canonical Model

(e) Canonical model warped into its live frame

(f) Model Normals

SUMMARY

- Visual SLAM is similar to SFM with Real-time constraints
- RT Features such as ORB
  - FAST corners
  - BRIEF Descriptors
- Place detection for Loop detection and Relocation
  - Bag of words