KinectFusion : Real – Time Dense Surface Mapping and Tracking

Richard A. Newcombe
Imperial College London

Shahram Izadi
Microsoft Research

Otmar Hilliges
Microsoft Research

David Molyneaux
Microsoft Research
Lancaster University

David Kim
Microsoft Research
Newcastle University

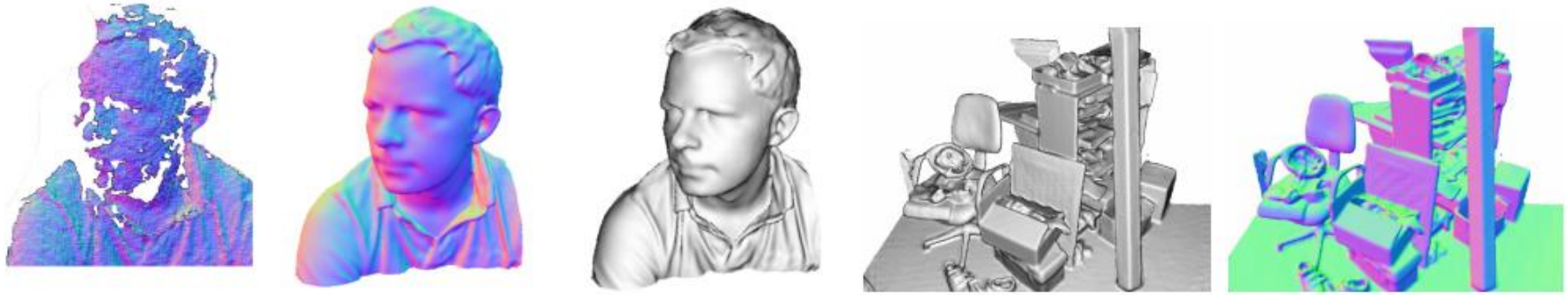
Andrew J. Davison
Imperial College London

Pushmeet Kohli
Microsoft Research

Jamie Shotton
Microsoft Research

Steve Hodges
Microsoft Research

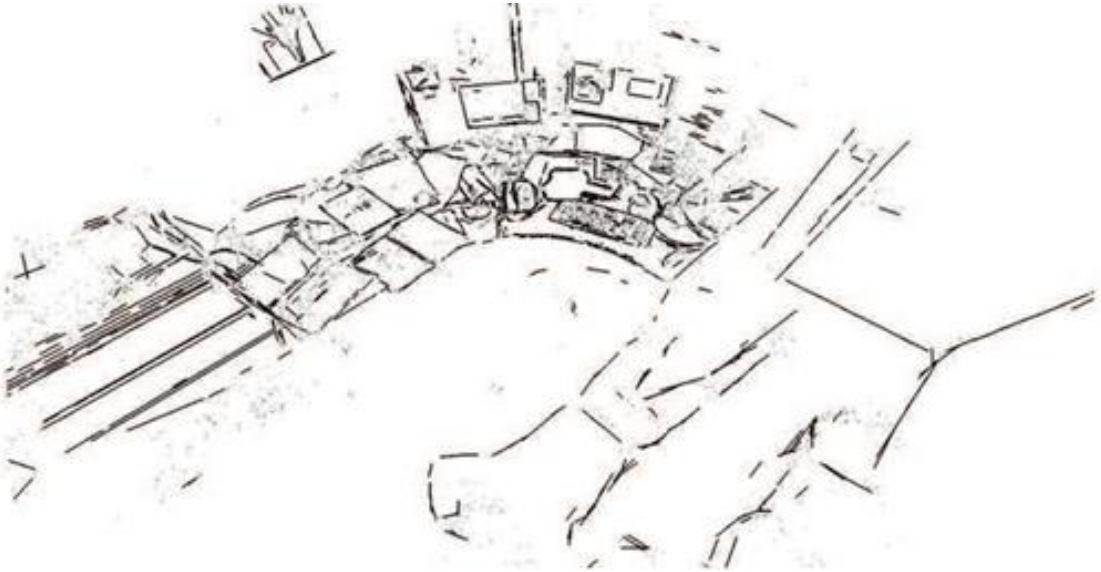
Andrew Fitzgibbon
Microsoft Research



RISHABH RAJ

RELATED WORK

- Handheld 3D Scanners
- SLAM
  - Monocular SLAM
    - Single handheld camera
    - Simultaneous tracking and mappin
  - PTAM
    - Sparse feature tracking
- Offline 3D reconstruction
  - Laser Range Scanners
  - SFM and MVS techniques that leverage online photos



MOTIVATION

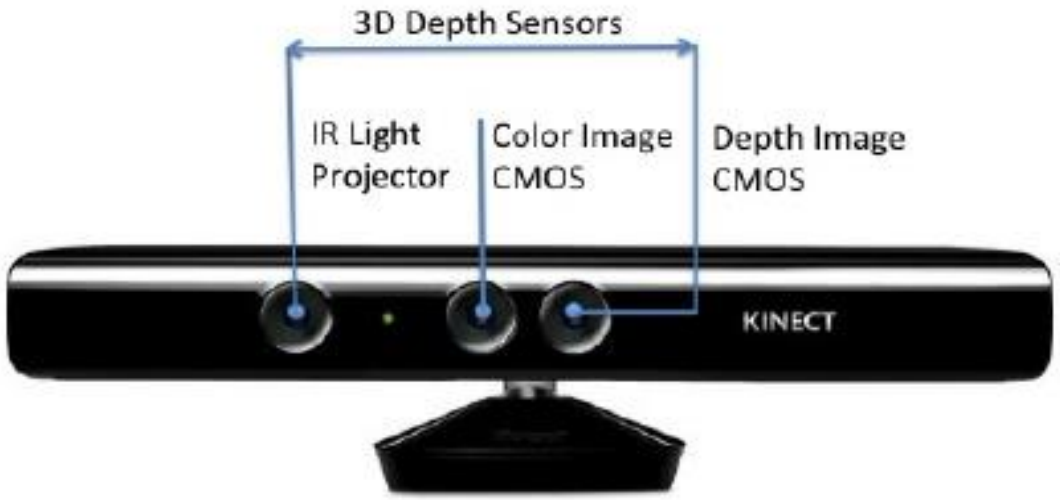
- Low cost Depth sensor
- Dense 3D reconstruction of live scene
- SLAM
- Infrastructure free
- Real Time
- Augmented and Mixed Reality

OUTLINE

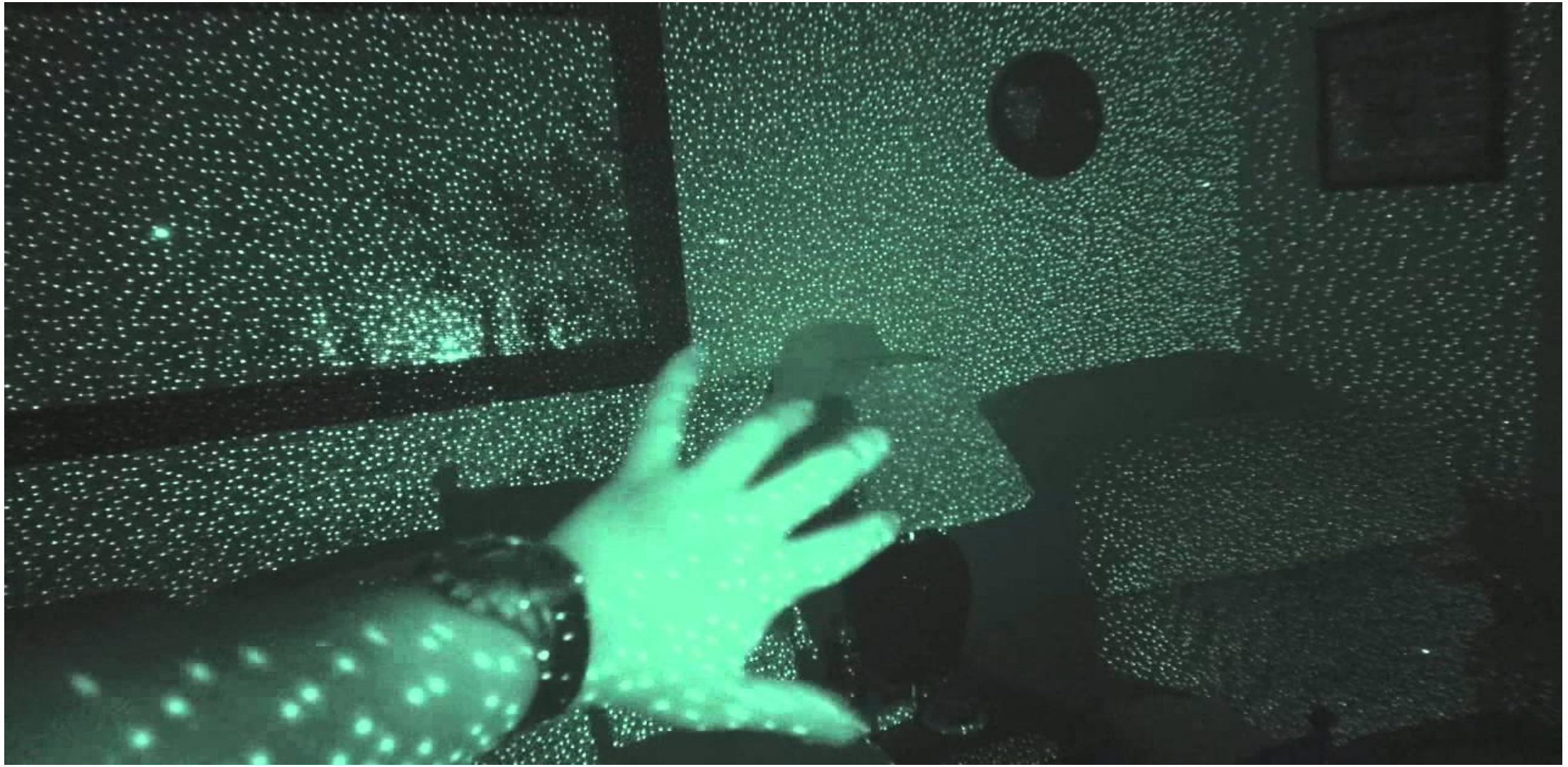
- System for accurate real – time tracking and mapping.
  - Indoors
  - Variable lighting conditions
  - Complex room sized scenes
- Dense Reconstruction
- GPU
- Microsoft Kinect
  - Low Cost
  - Depth Sensor

KINECT

- IR laser projects a speckled light pattern
- Stereo matching algorithm for depth sensing
- Producing a 640 × 480 depth image at 30fps



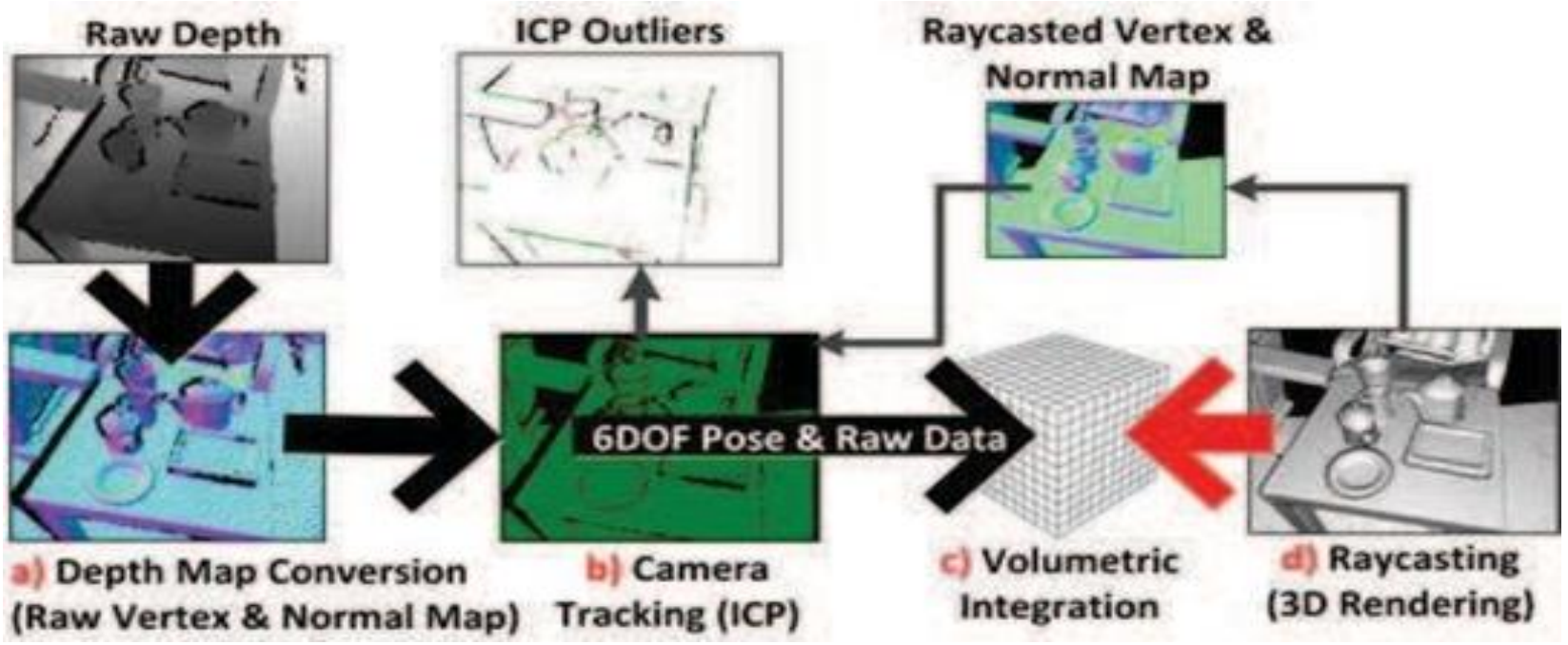
KINECT



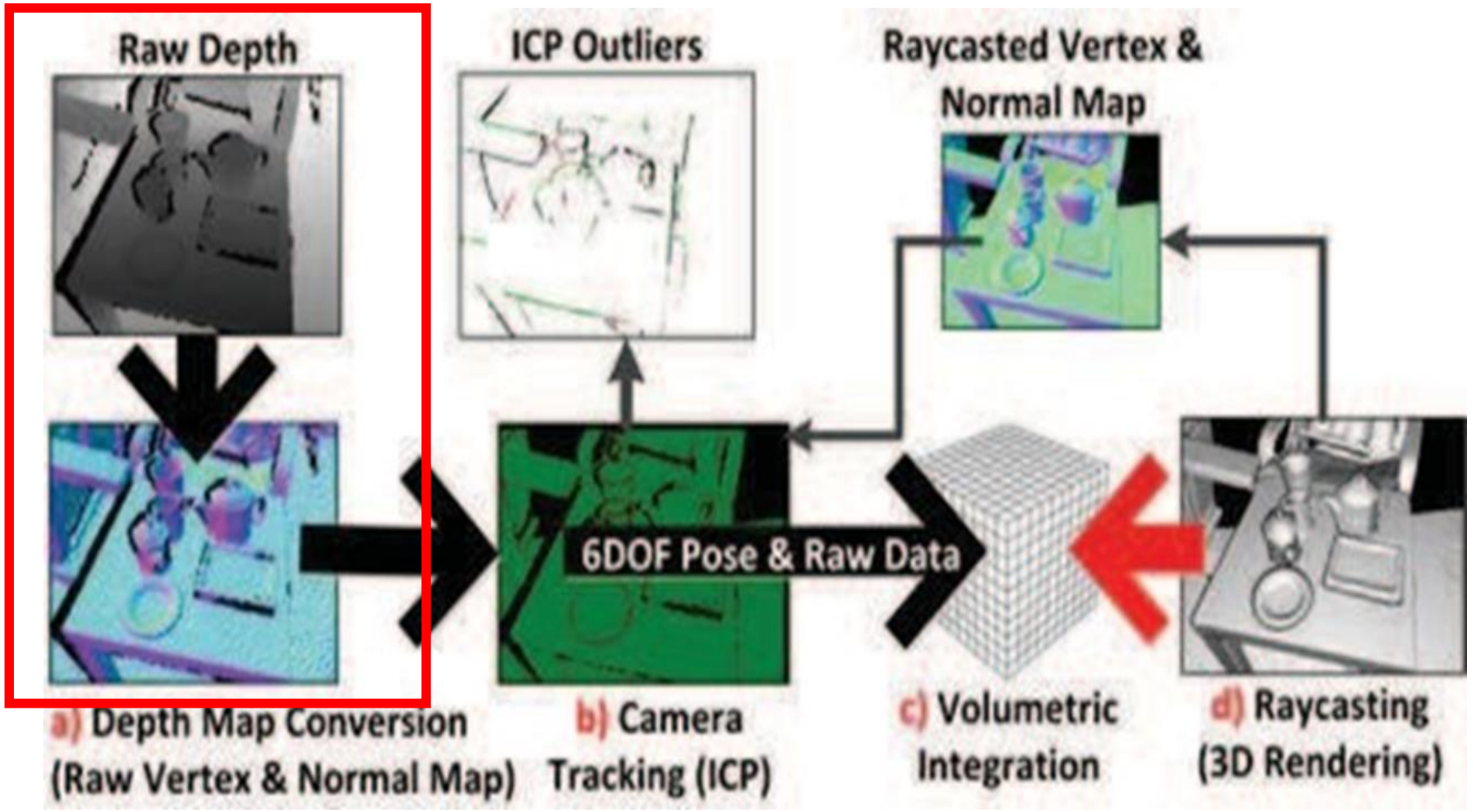
KINECT

APPROACH

- Two simple interleaved components
  - Building a dense surface model from a set of depth frames with estimated camera poses
  - Given a dense surface model, estimate the current camera pose by aligning the depth frame in the dense model.



DEPTH MAP CONVERSION



VERTEX MAP : 3D DENSE POINT CLOUDS

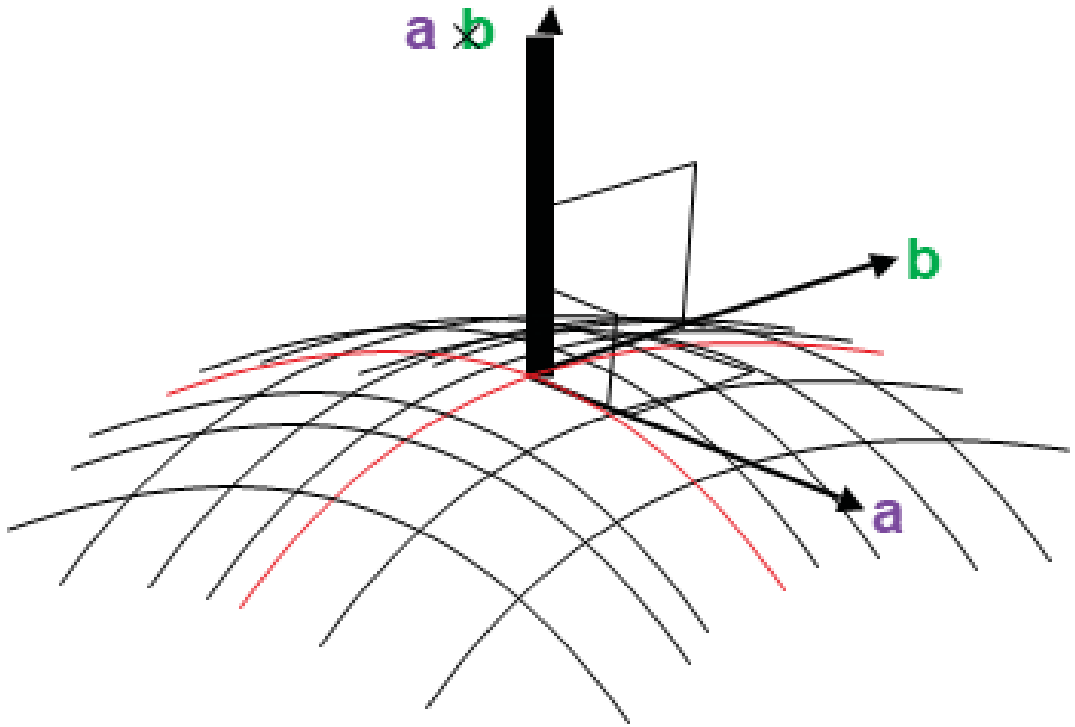
- Filter raw depth data with a bilateral filter
- The depth map at time *k* provides a 3D point measurement at each pixel, a vertex map V_k .
- Transformation of the 3*D* point from the camera to world frame is $v_w = T_{w,k} v_k$

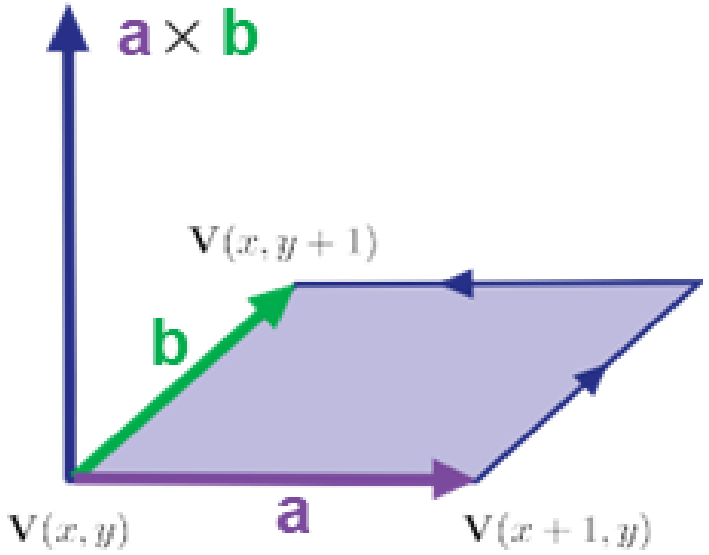
$$v = K^{-1}[x, y, 1]^\top D(x, y)$$

$$K \equiv \begin{pmatrix} f_0 & 0 & p_0 \\ 0 & f_1 & p_1 \\ 0 & 0 & 1 \end{pmatrix}$$

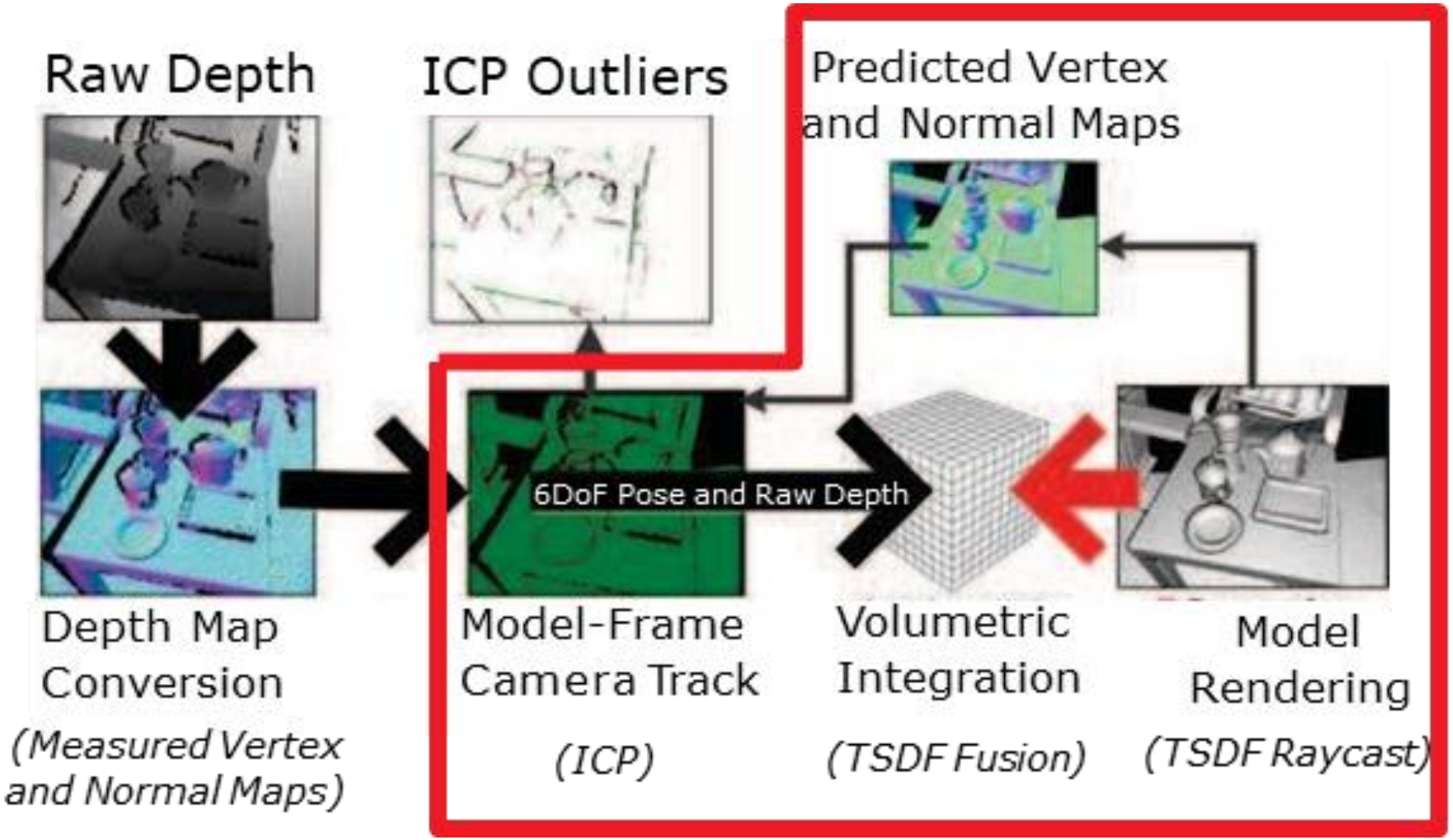
SURFACE NORMALS

We can estimate the surface normal from neighbouring pairs of 3D points by exploiting the regular grid structure





CAMERA TRACKING



CAMERA TRACKING

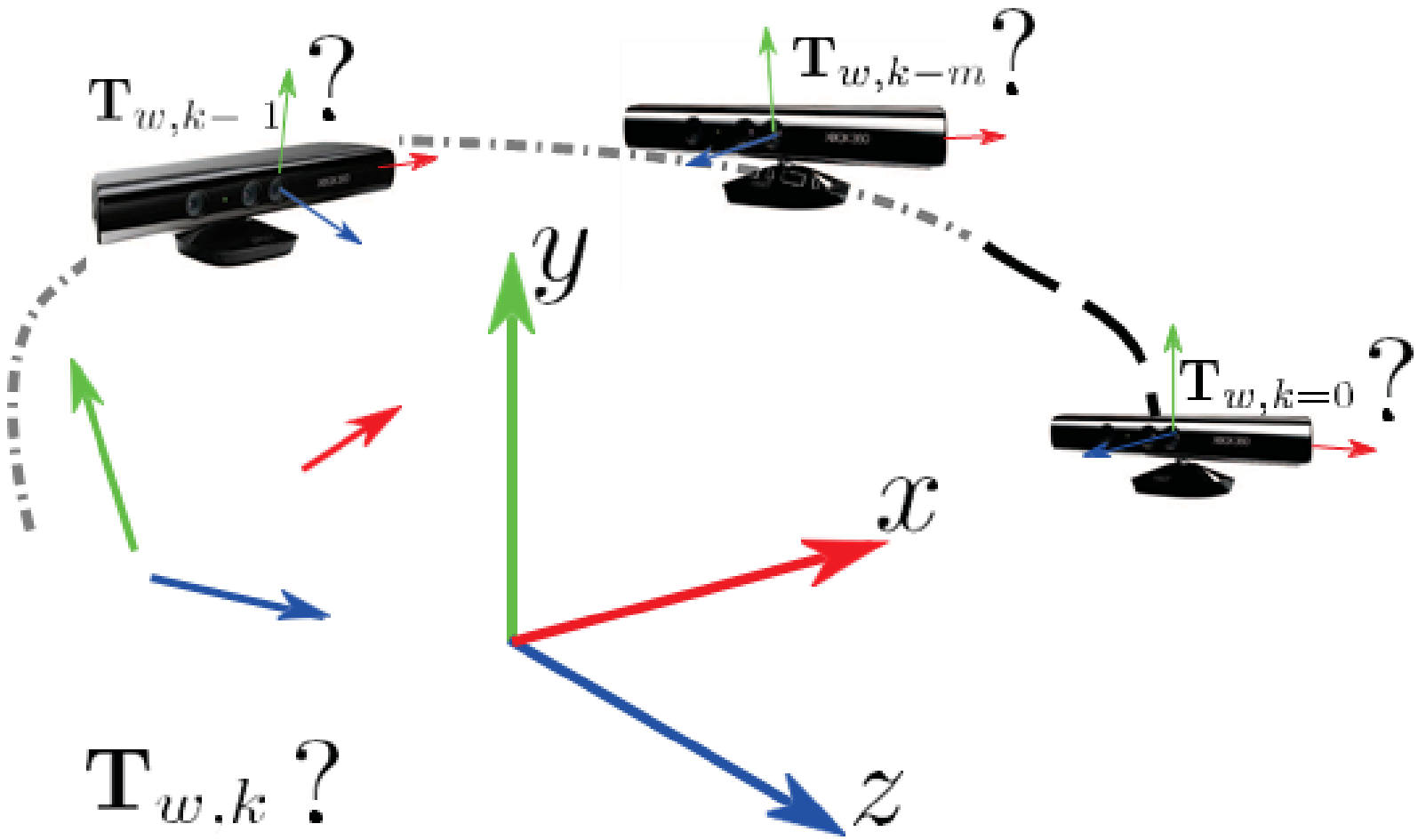
- For frame *k* the pose of the camera is given by the six degree of freedom rigid body transform

$$\mathbf{T}_{w,k} = \begin{bmatrix} \mathbf{R}_{w,k} & \mathbf{t}_{w,k} \\ \mathbf{0}^\top & 1 \end{bmatrix} \in \mathbb{SE}_3$$

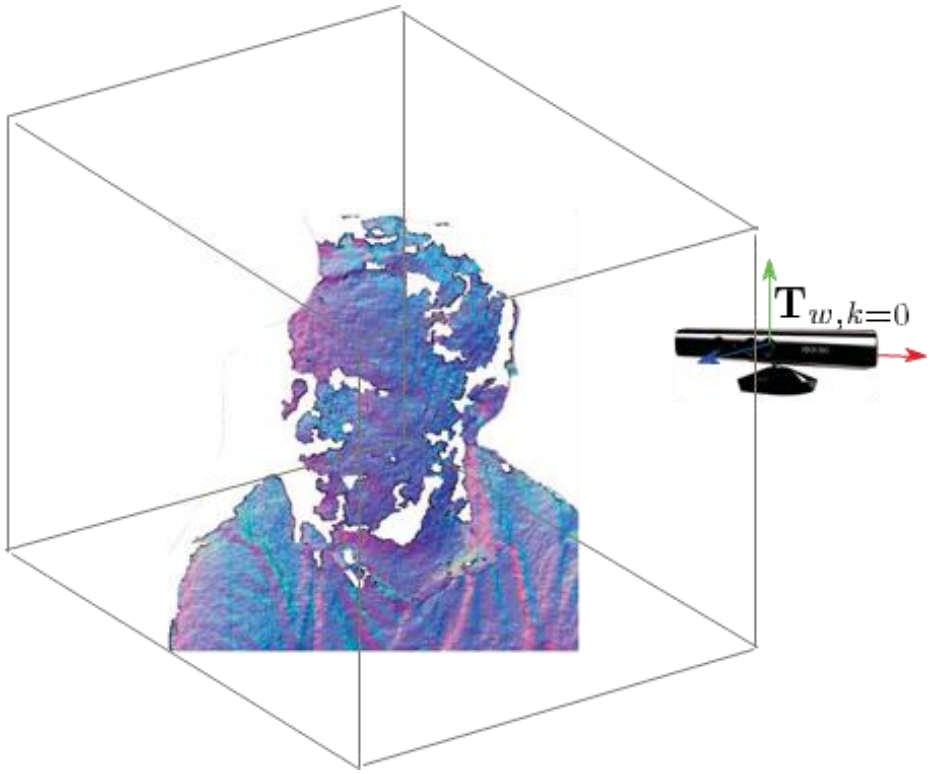
$$\mathbb{SE}_3 := \{\mathbf{R}, \mathbf{t} \mid \mathbf{R} \in \mathbb{SO}_3, \mathbf{t} \in \mathbb{R}^3\}$$

- ICP Algorithm is used for tracking the camera.

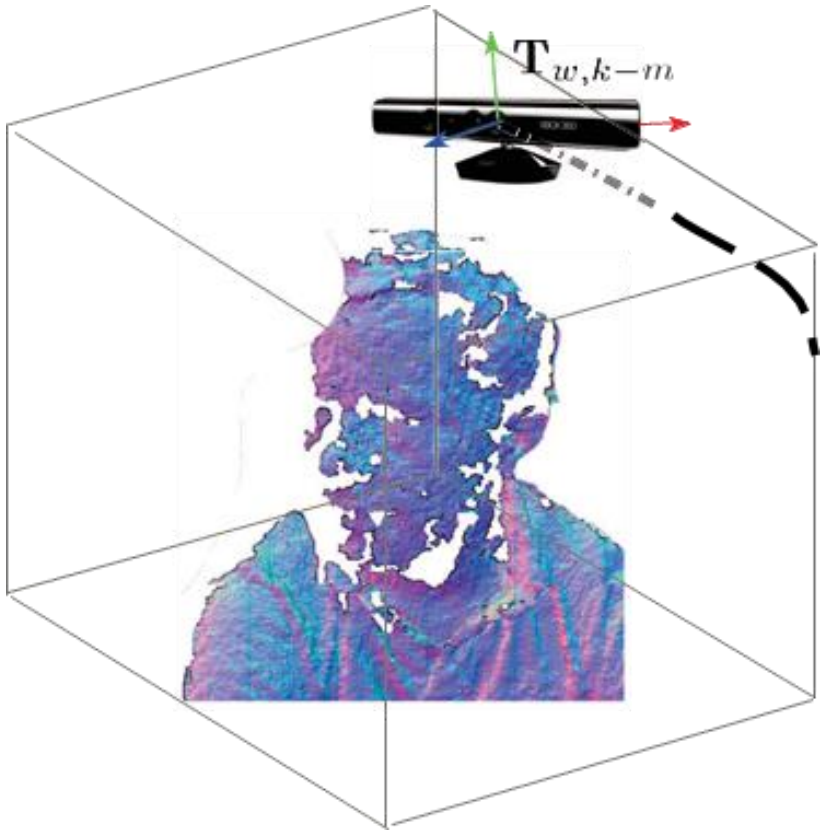
CAMERA TRACKING



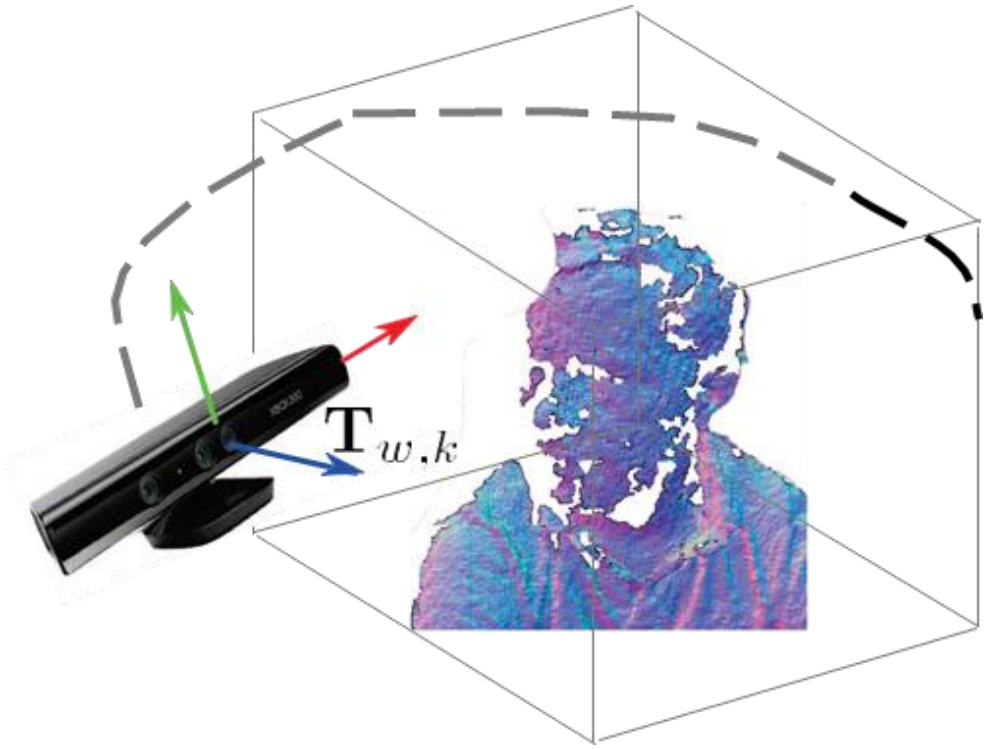
Knowing camera motion, enables model reconstruction



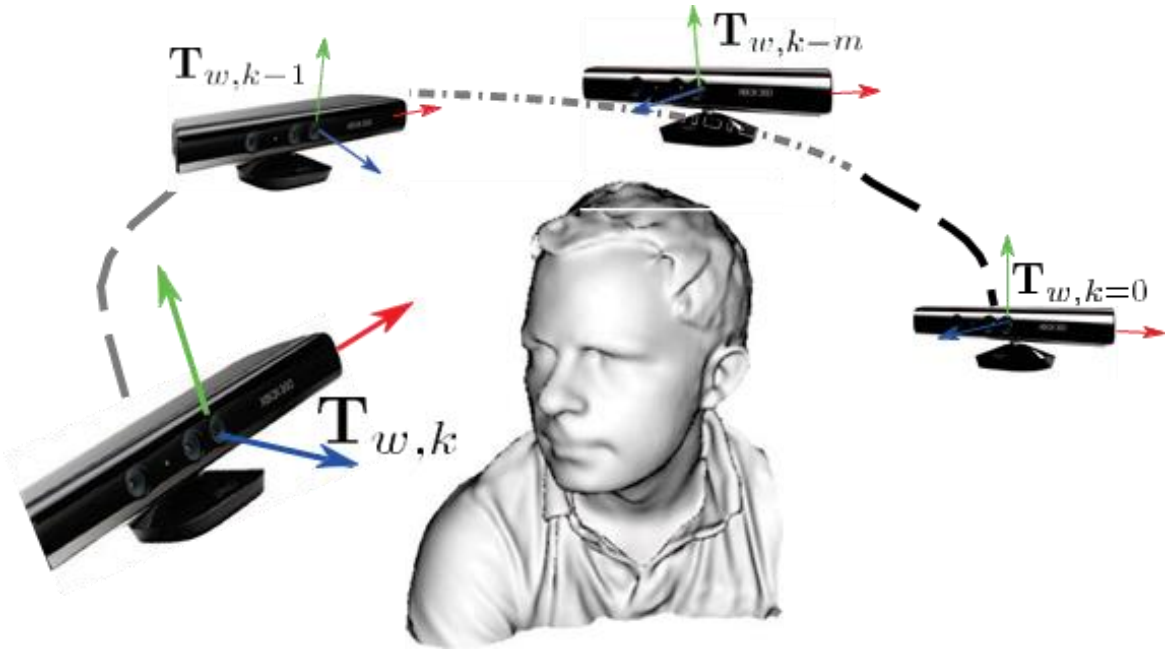
Knowing camera motion…



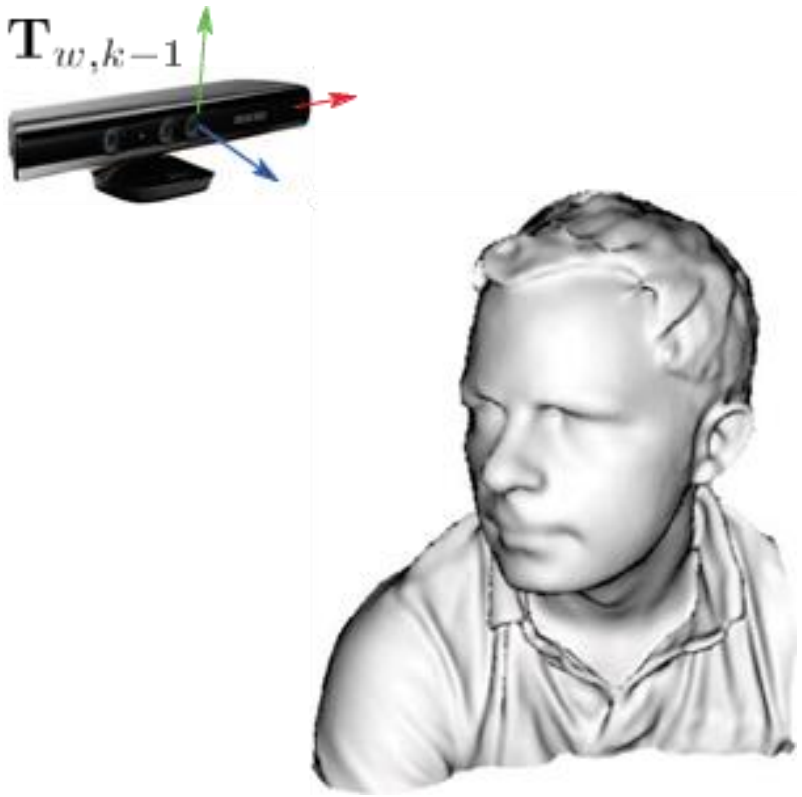
Knowing camera motion…



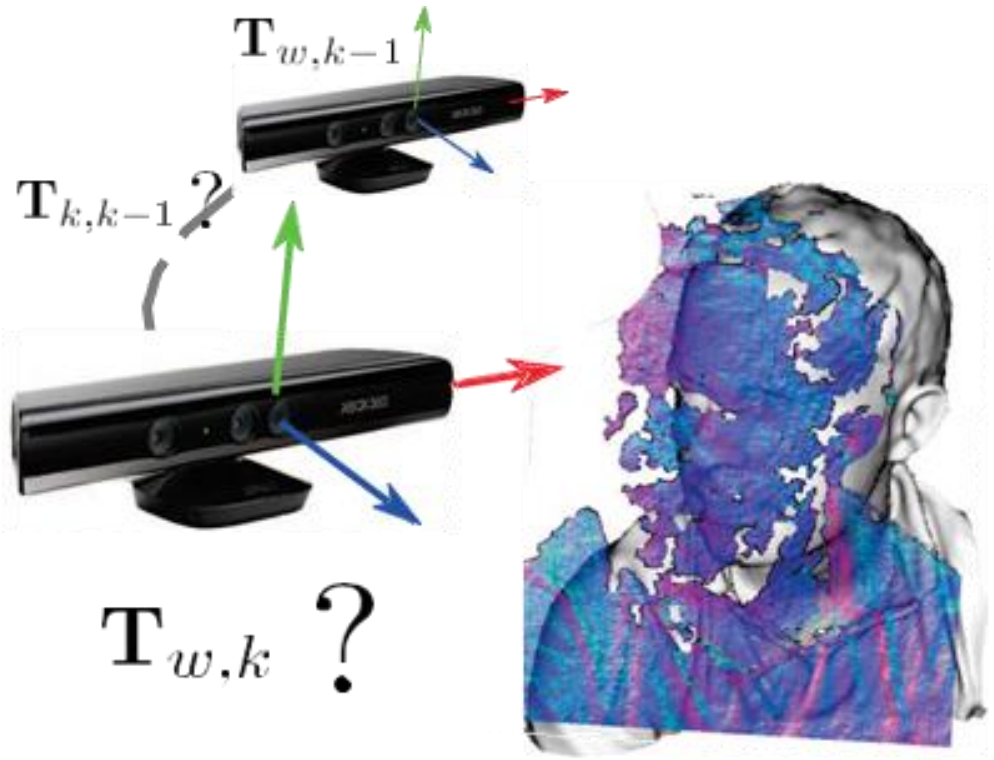
… enables measurement fusion



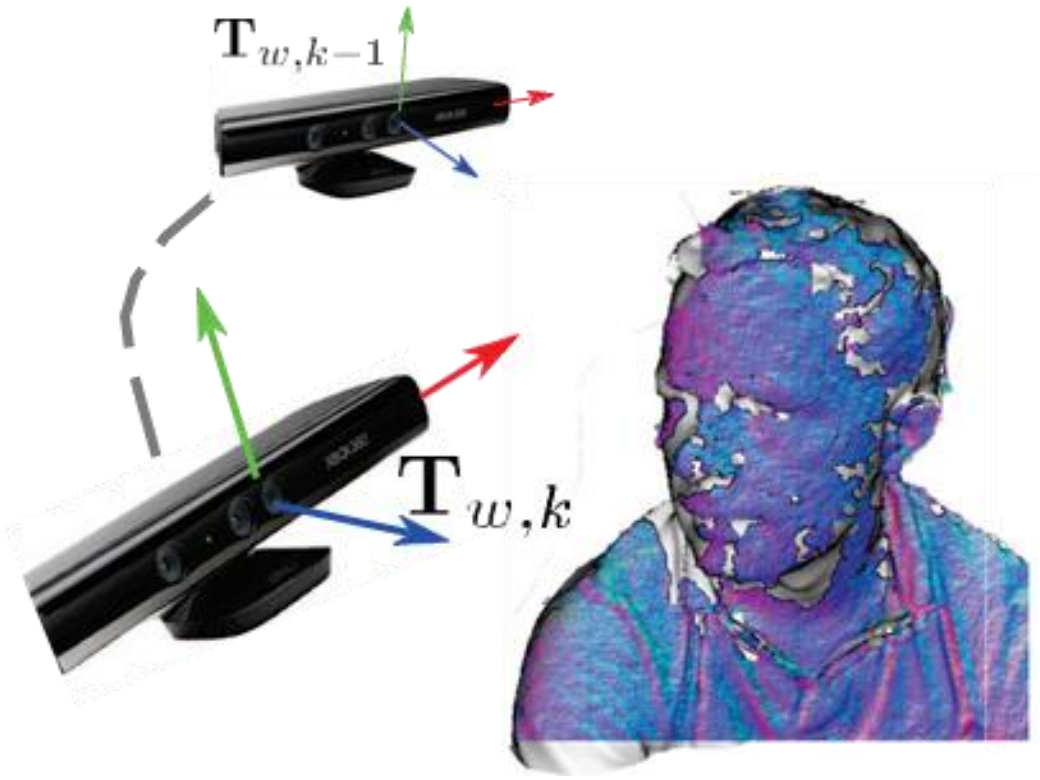
…also, given a known model…



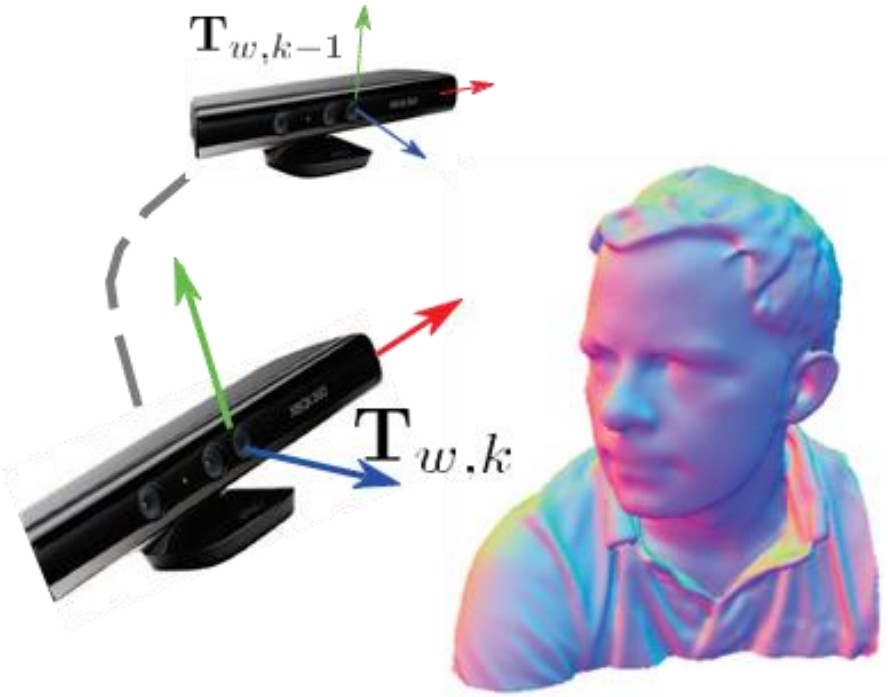
…can align a new surface measurement



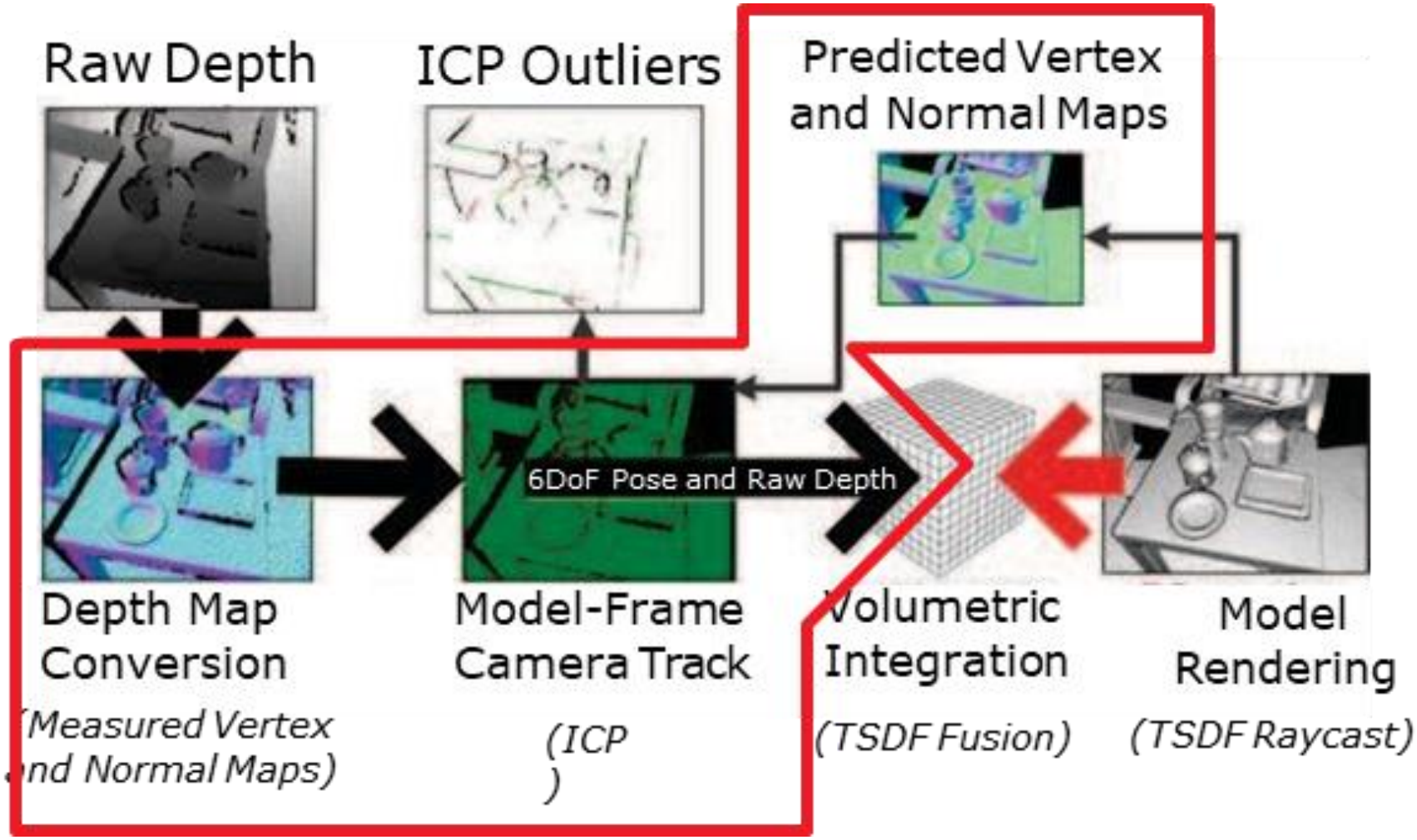
…minimizing the predicted surface measurement error



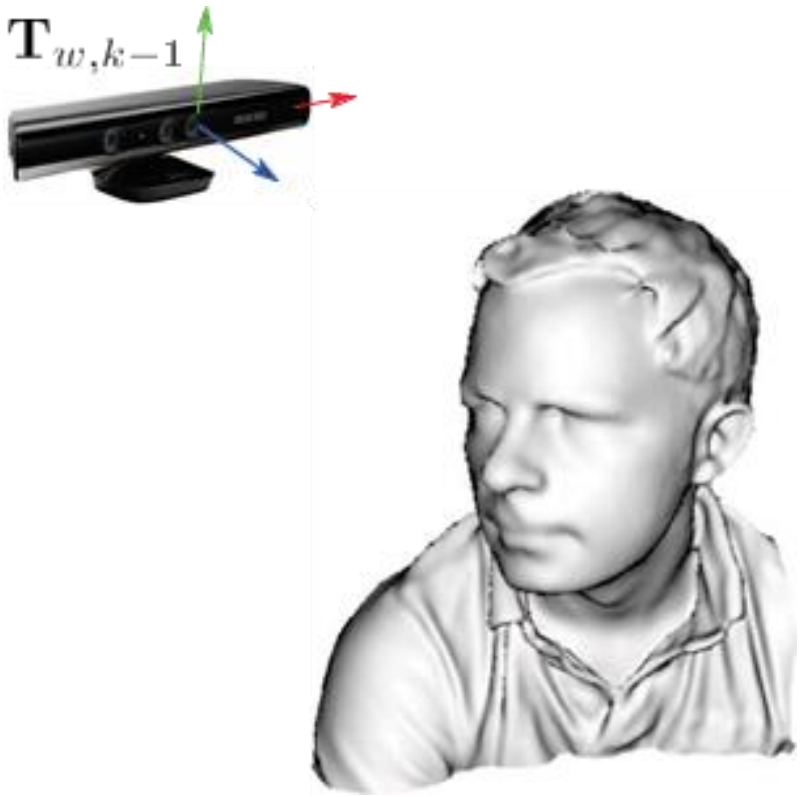
…gives a best current pose estimate, enabling fusion



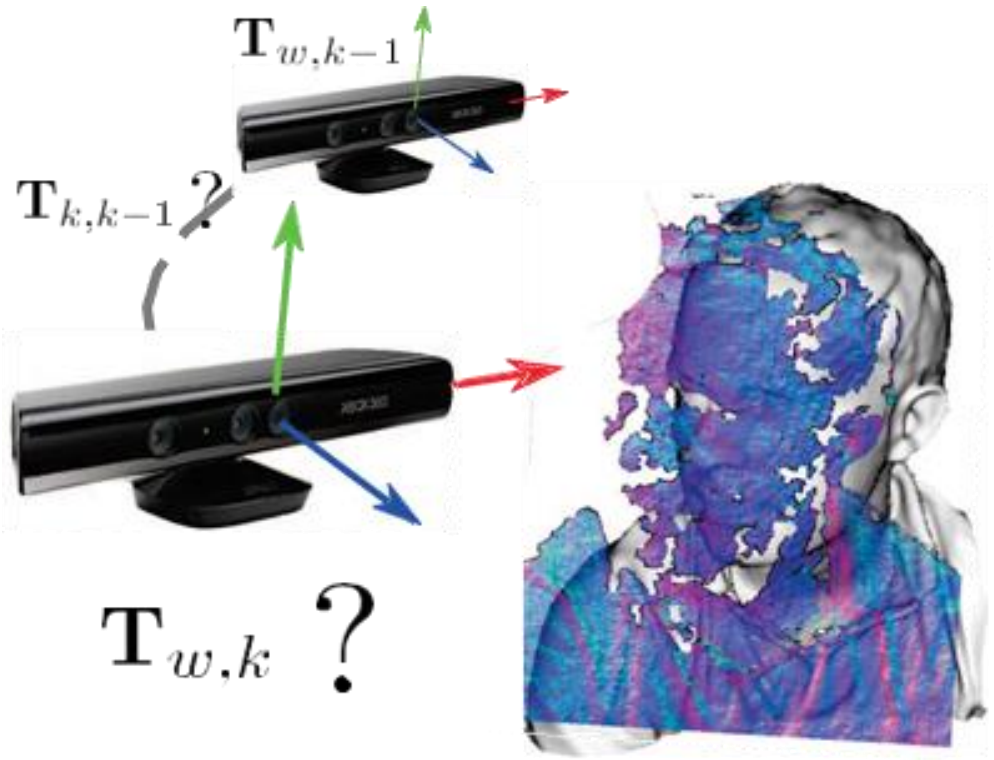
DENSE MAP TO DENSE SURFACE ALINGMENT



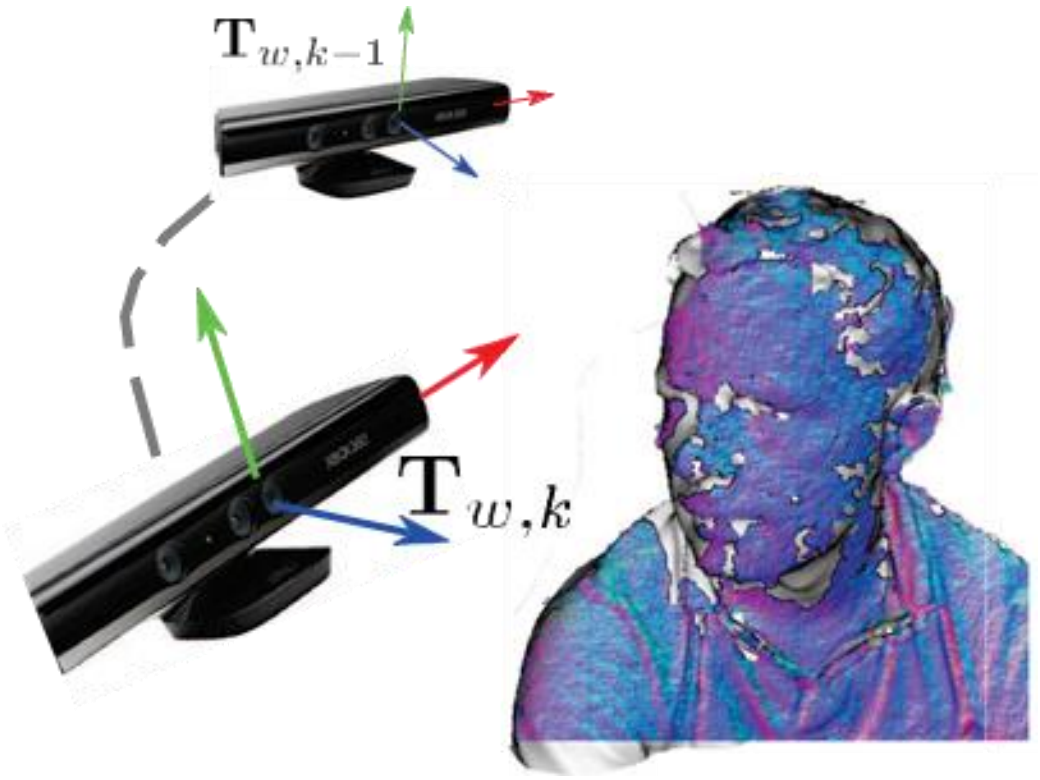
Given a known, partially complete surface model



To estimate a new camera frame 6DoF pose

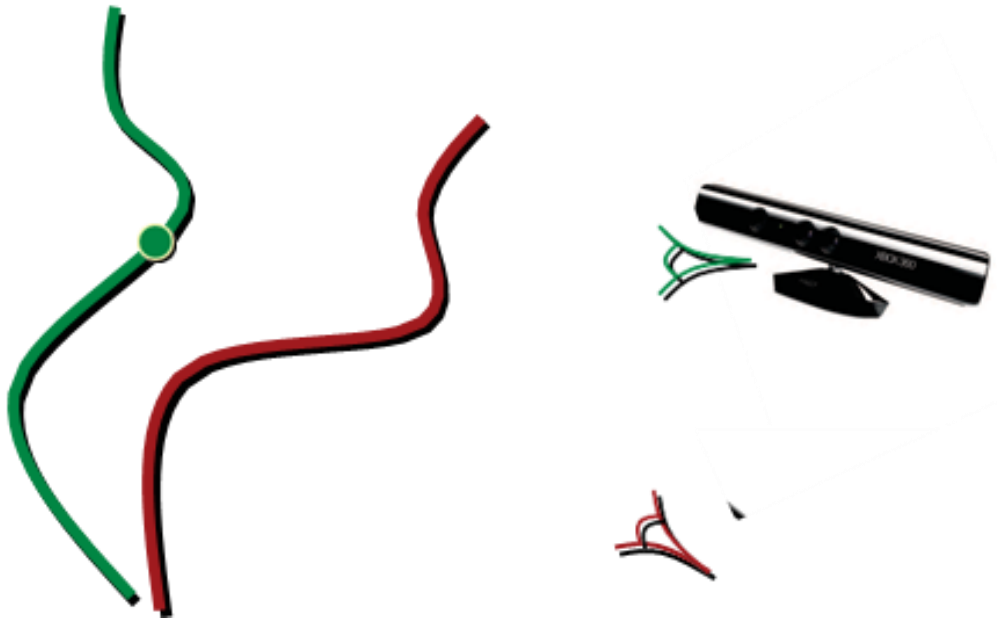


Which is equivalent to finding the pose that aligns the depth map data onto the current model

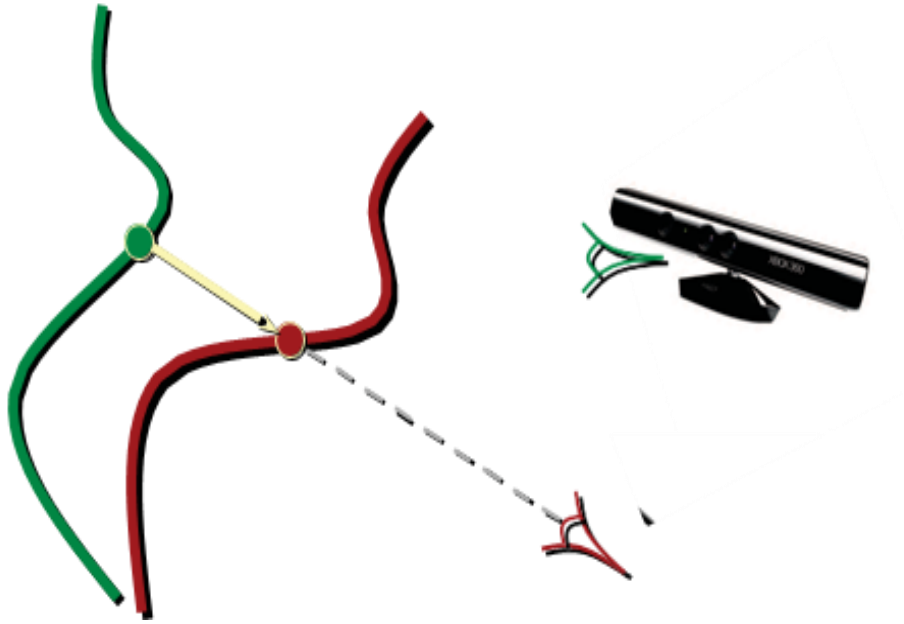


PROJECTIVE DATA ASSOCIATION

Select a point from the reference surface

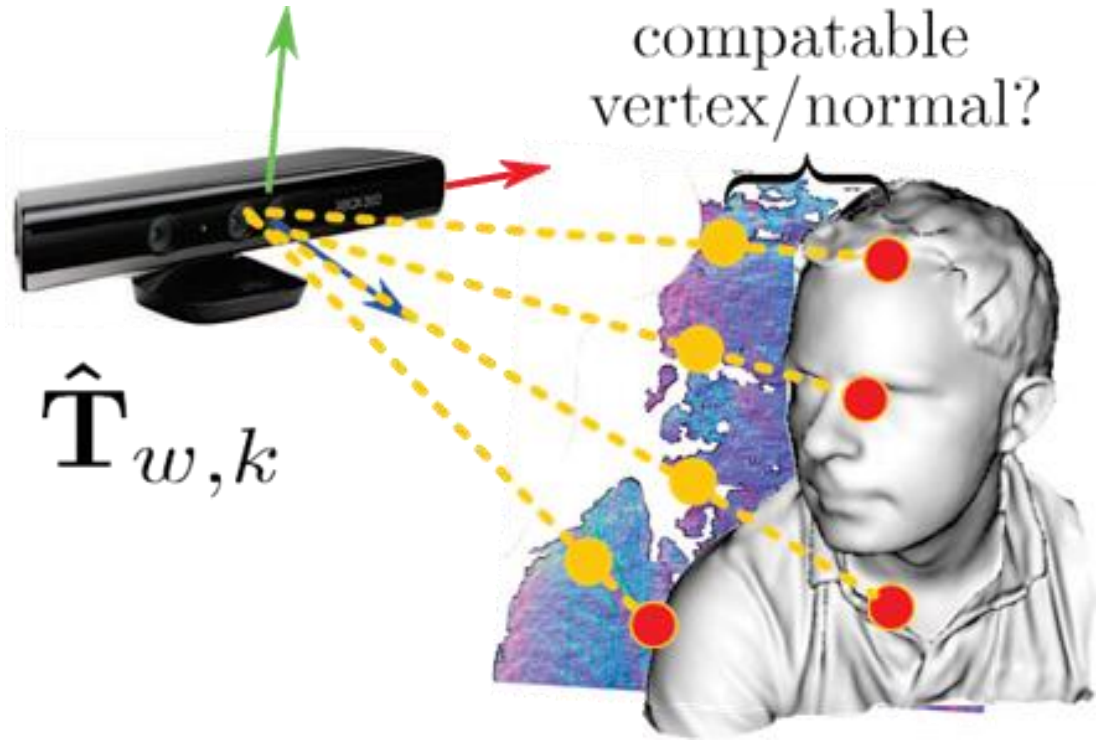


Project the into the frame of the second surface measurement

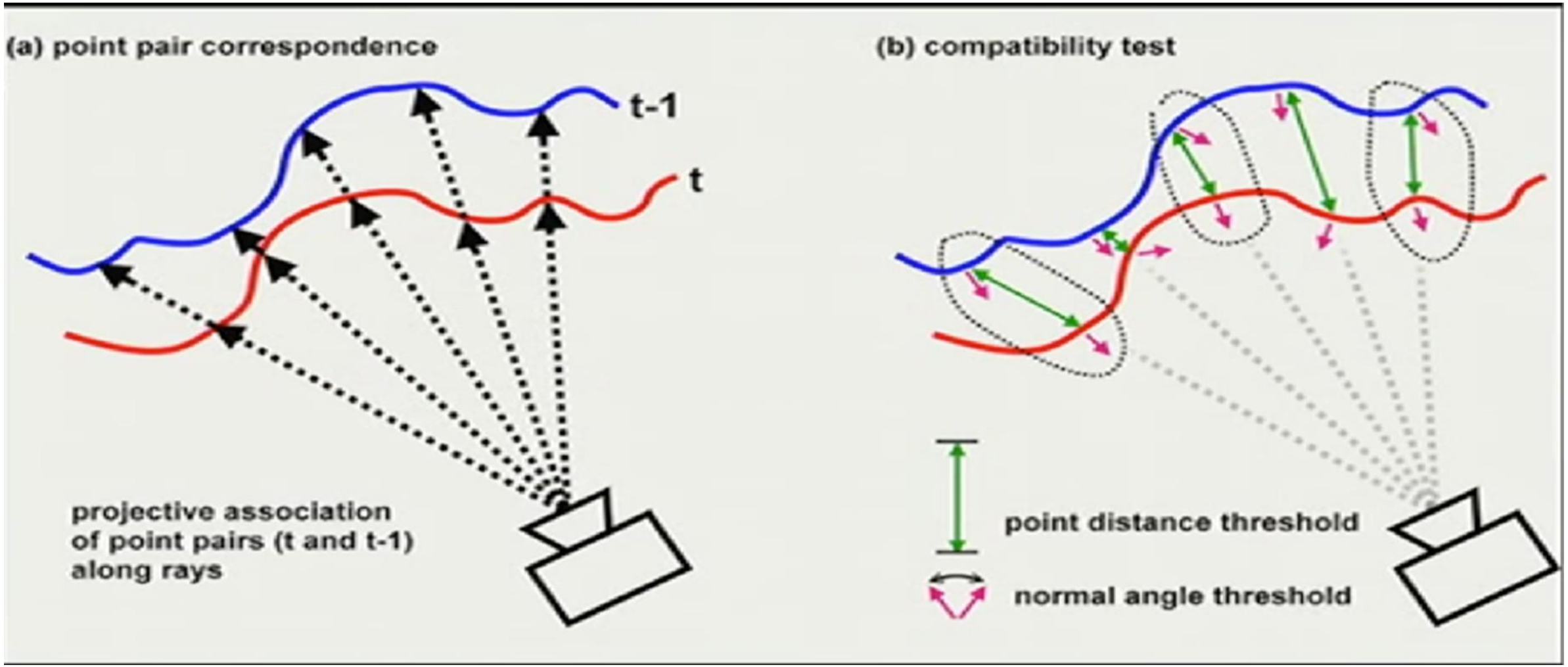


DENSE PROJECTIVE DATA ASSOCIATION

- Predicts a depth map by raycasting the current surface model given camera pose
- Accept match only if surface normal are similar and if point distance is not too large



DATA ASSOCIATION



POINT PLANE ICP

- Given data-association between a model and a live depth frame, estimate the 6DoF transform that aligns the surfaces
- Vertex $v_k(u)$ in pixel u is data-associated with the global model predicted vertex $v_w(u')$ at pixel u' with normal n_w
- $\psi(.)$ is a penalty function, typically chosen as the squared distance function

Point-Plane Distance metric (Y. Chen and G. Medioni, 1992)

Point-plane error for a given transform $T_{w,k} \in \mathbb{SE}(3)$, over all associated points:

$$E_c(T_{w,k}) = \sum_{u \in \Omega} \psi\left(e(u, T_{w,k})\right) \ ,$$

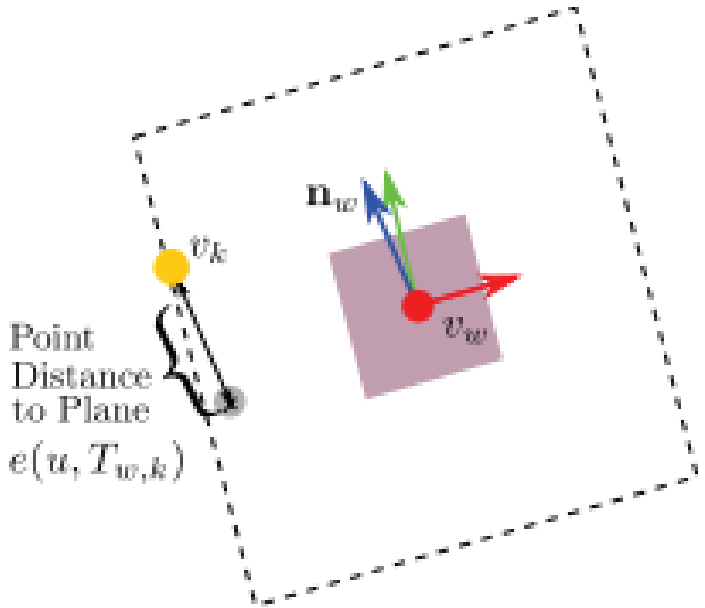
$$e(u, T_{w,k}) = \mathbf{n}_w(u')^\top \left(T_{w,k} v_k(u) - v_w(u')\right) \ .$$

POINT PLANE ICP

- Point-plane metric allows surfaces to slide over each other and compliments the projective data-association method

Point-Plane Distance metric (Y. Chen and G. Medioni, 1992)

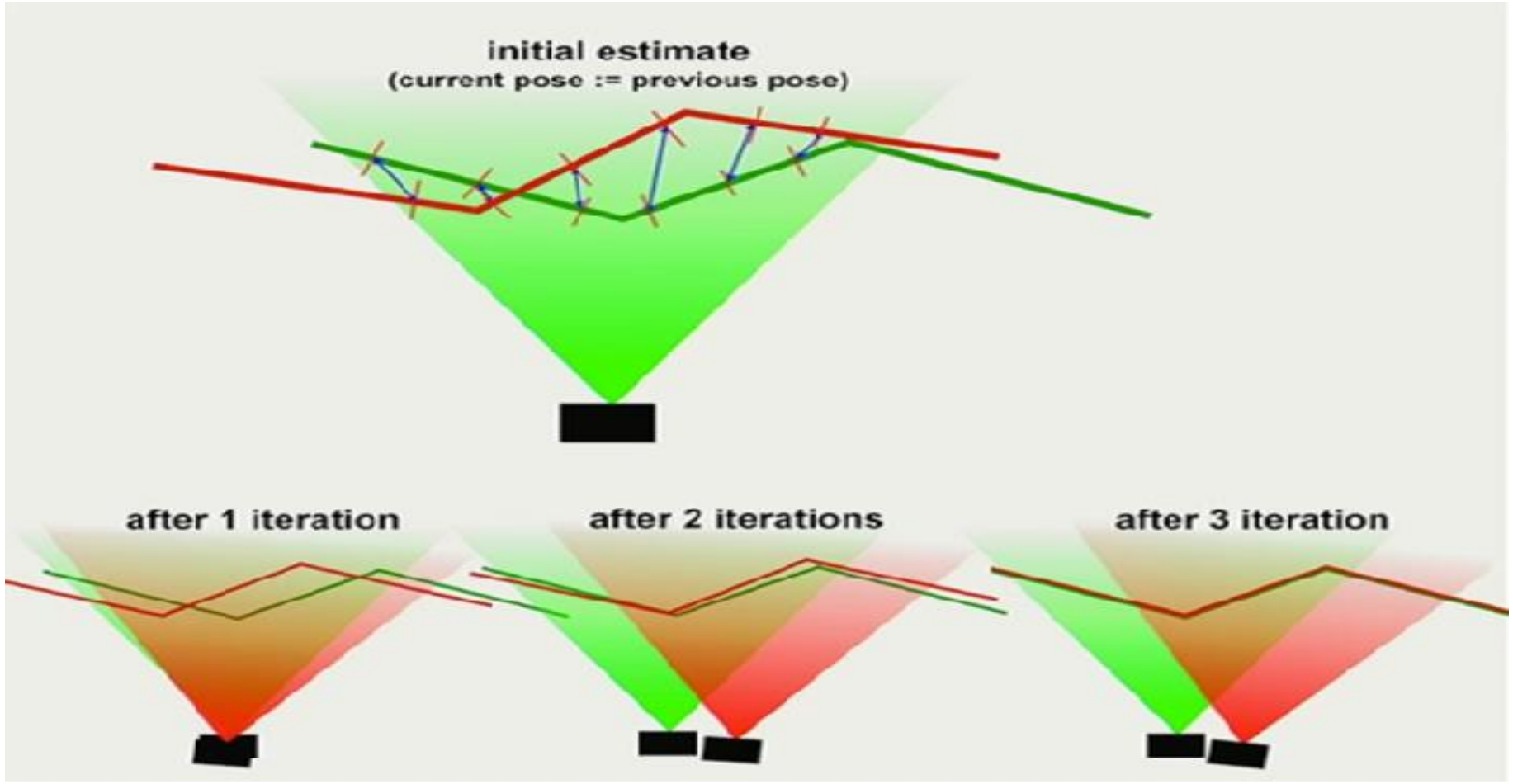
$$e(u, T_{w,k}) = \mathbf{n}_w(u')^\top (T_{w,k} v_k(u) - v_w(u'))$$



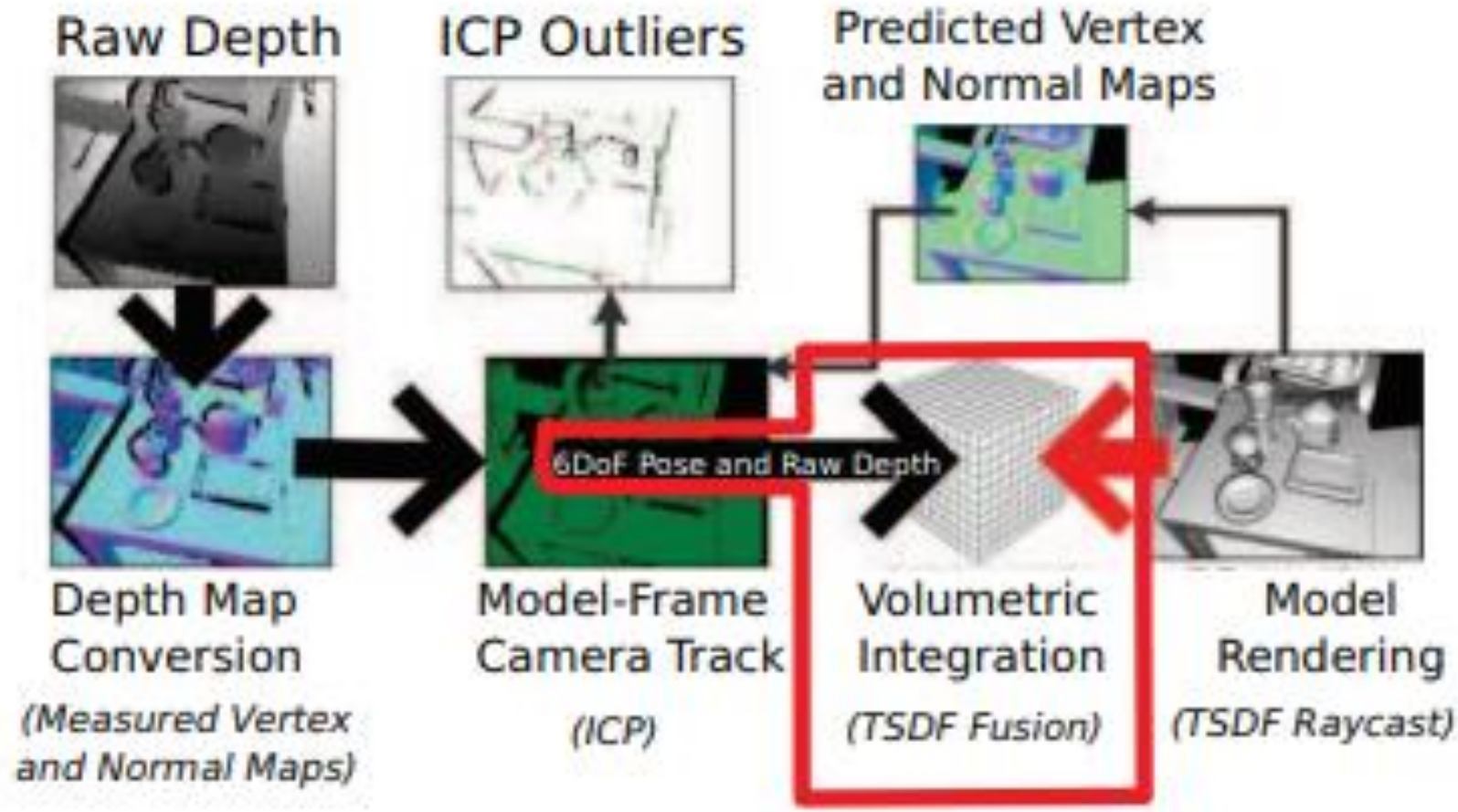
FAST ICP

- The combination of both projective data-association and the point-plane metric is called Fast ICP:
  - Given current point correspondences, minimize $E_c(T_{w,k})$
  - Use new estimate of $T_{w,k}$ and update correspondences

CAMERA TRACKING

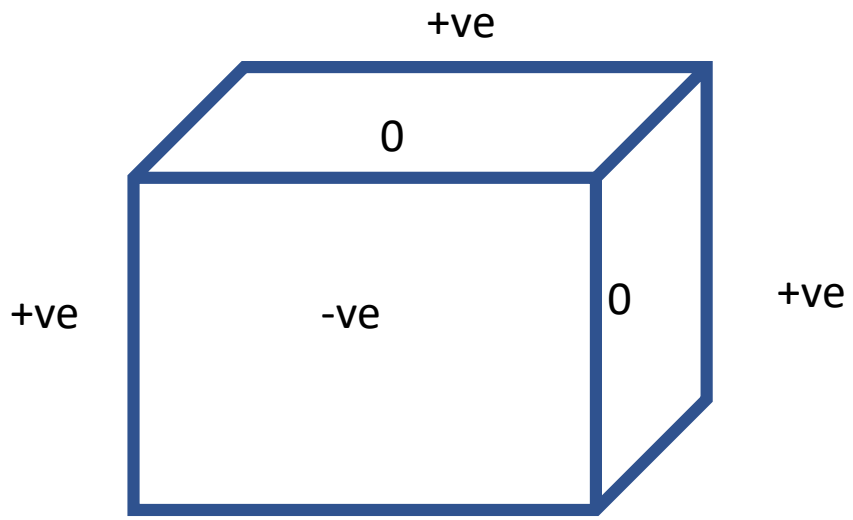


SURFACE RECONSTRUCTION

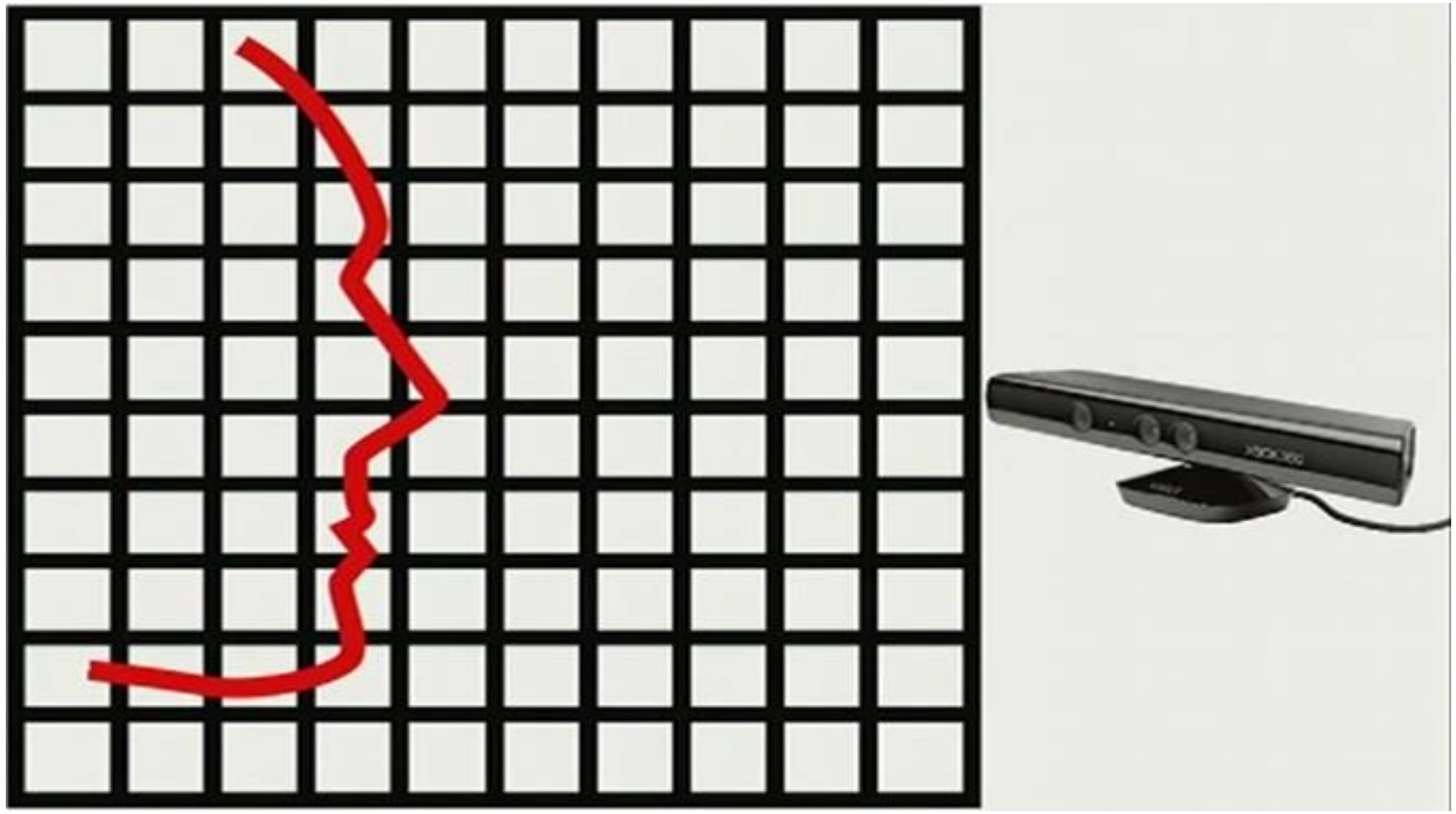


VOLUMETRIC INTEGRATION

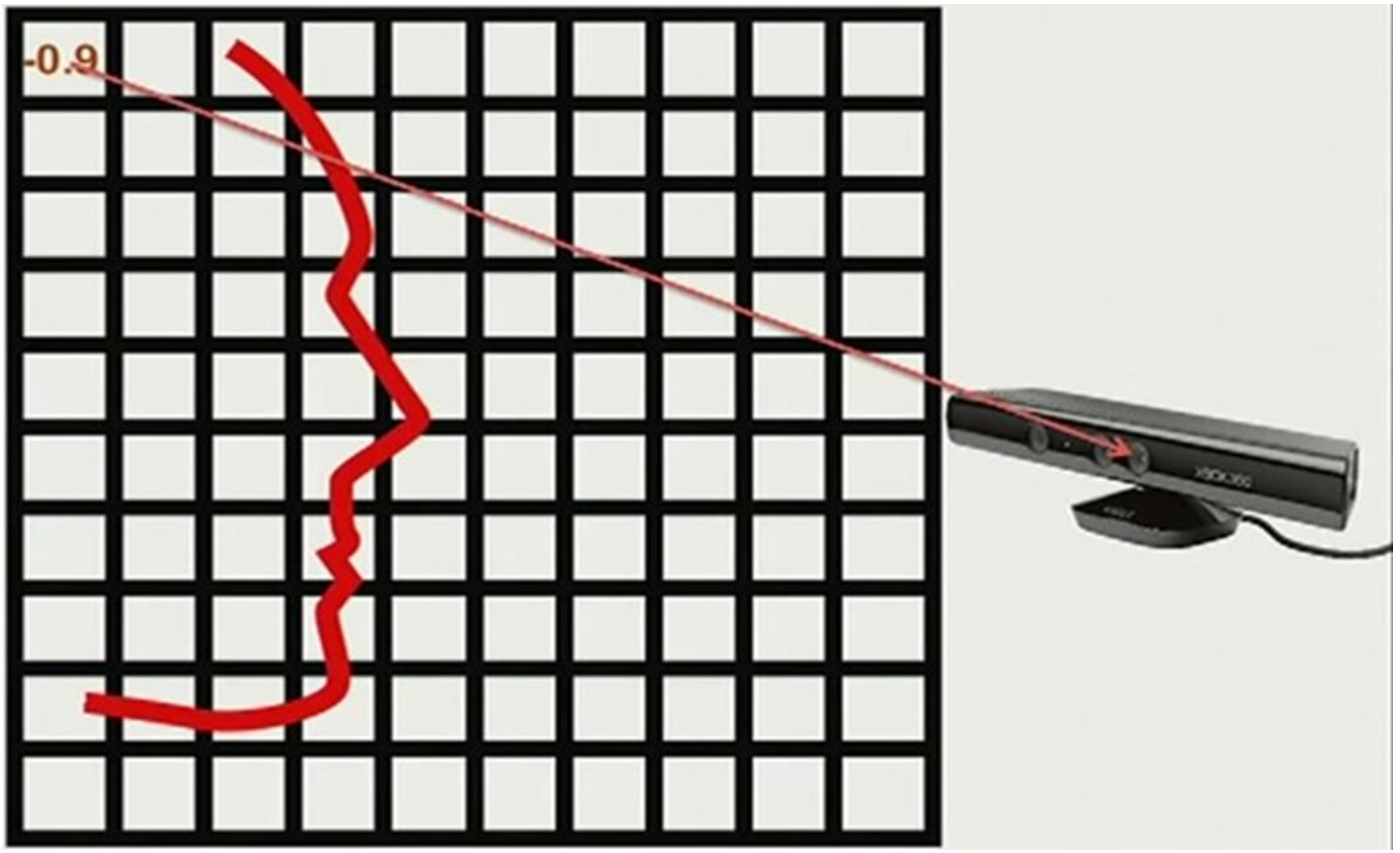
- Volume is subdivided uniformly into a 3D grid of voxels
- Implicit surfaces modeled with Truncated Signed Distance Function (TSDF)
- Voxels within a certain distance to a probable surface store signed (+/-) distance values to the surface
- Changing topology is costly and complicated for parametric or explicit mesh representations.



TSDF

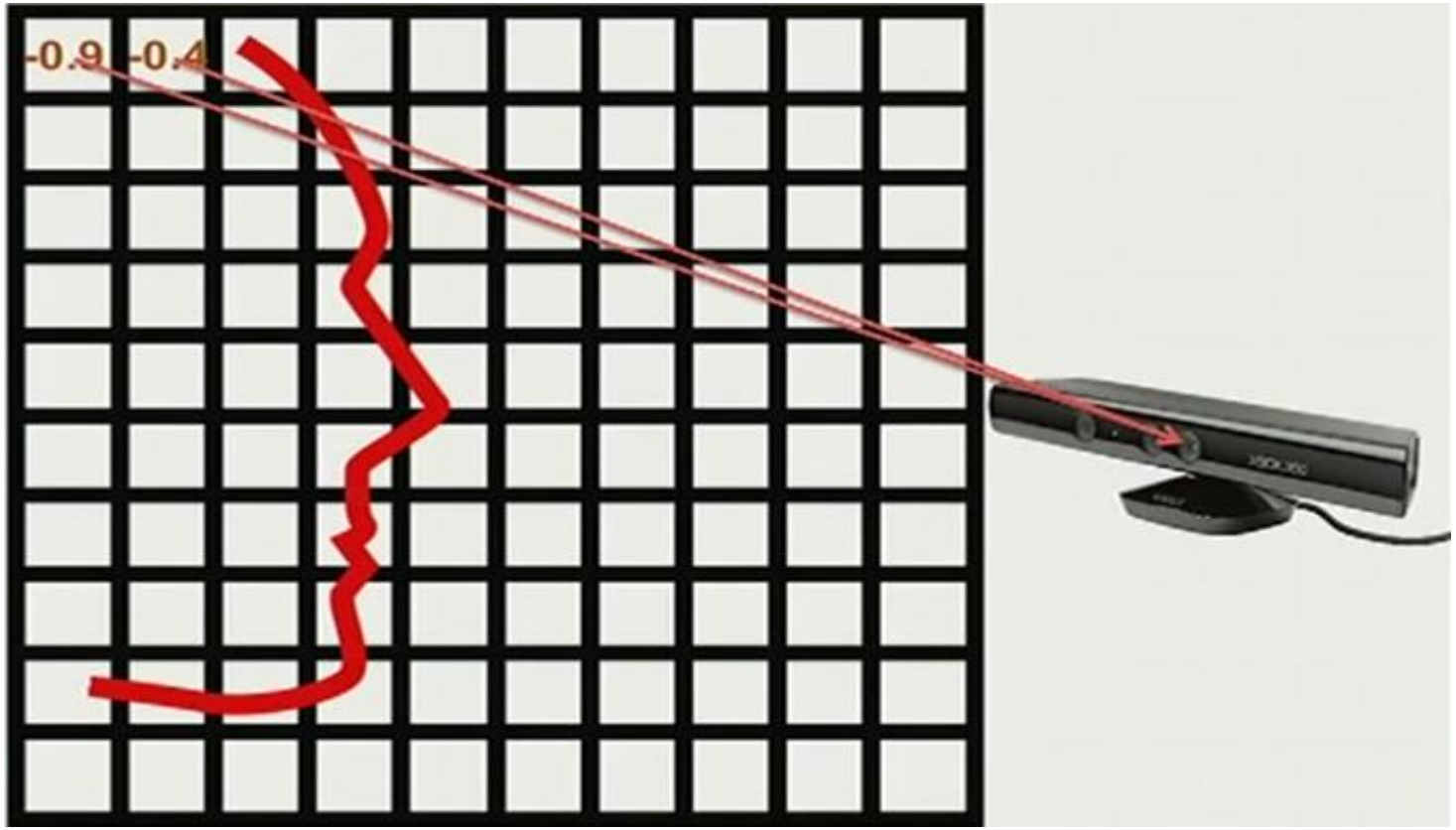


TSDF



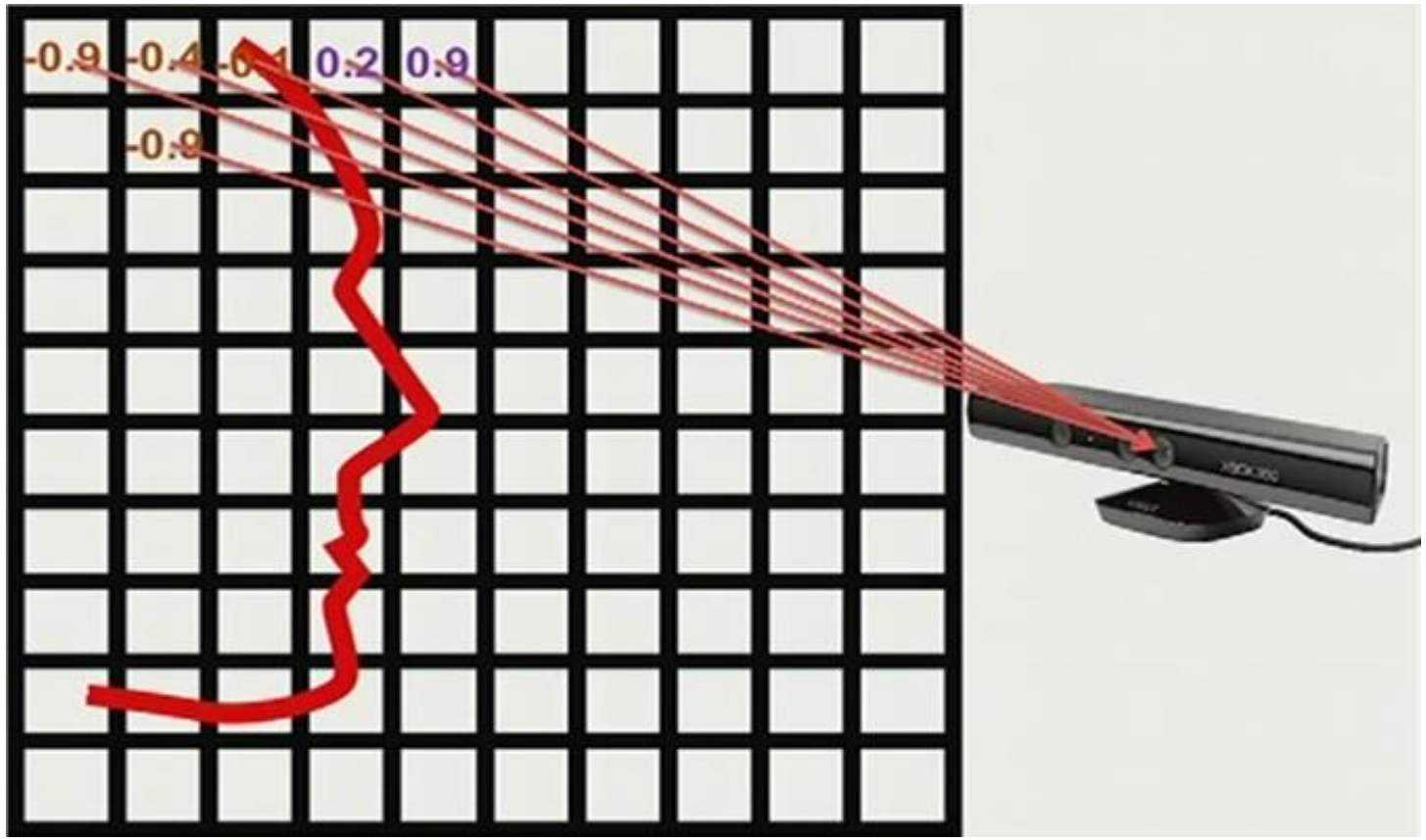
d = [pixel depth] – [distance from sensor to voxel]

TSDF



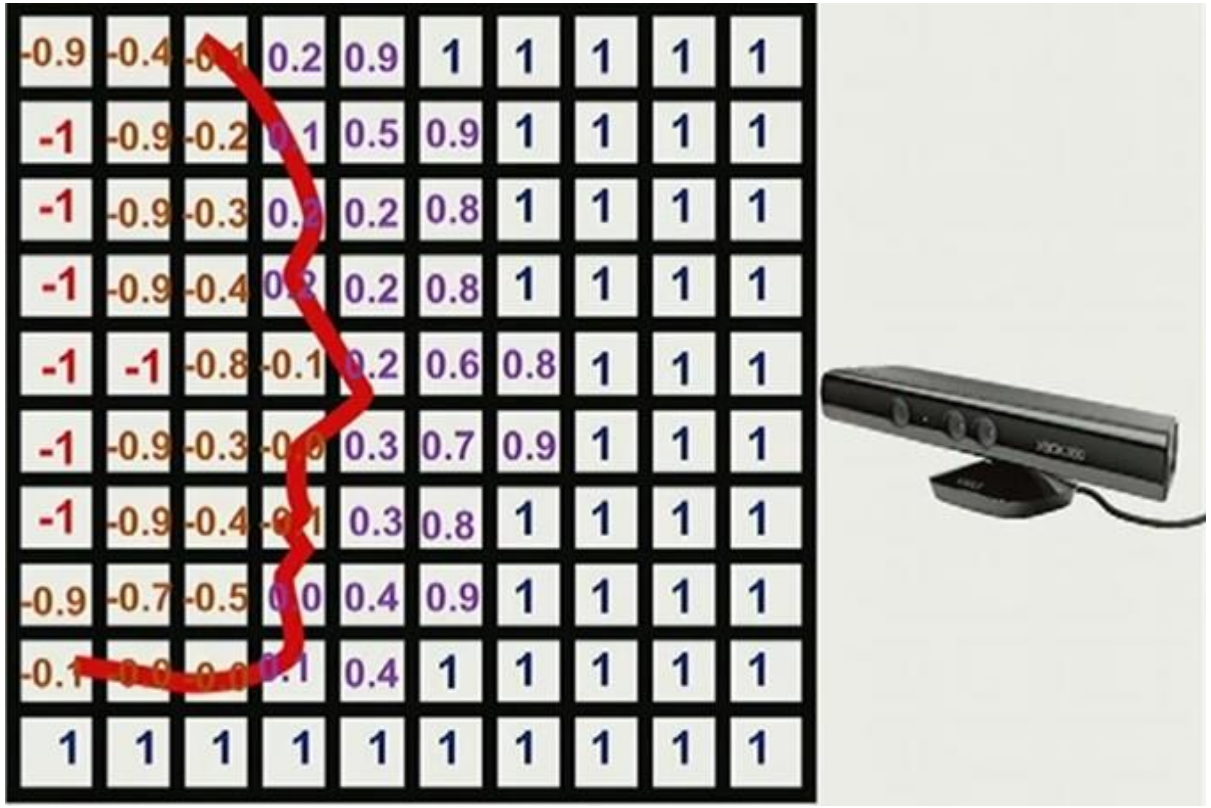
d = [pixel depth] – [distance from sensor to voxel]

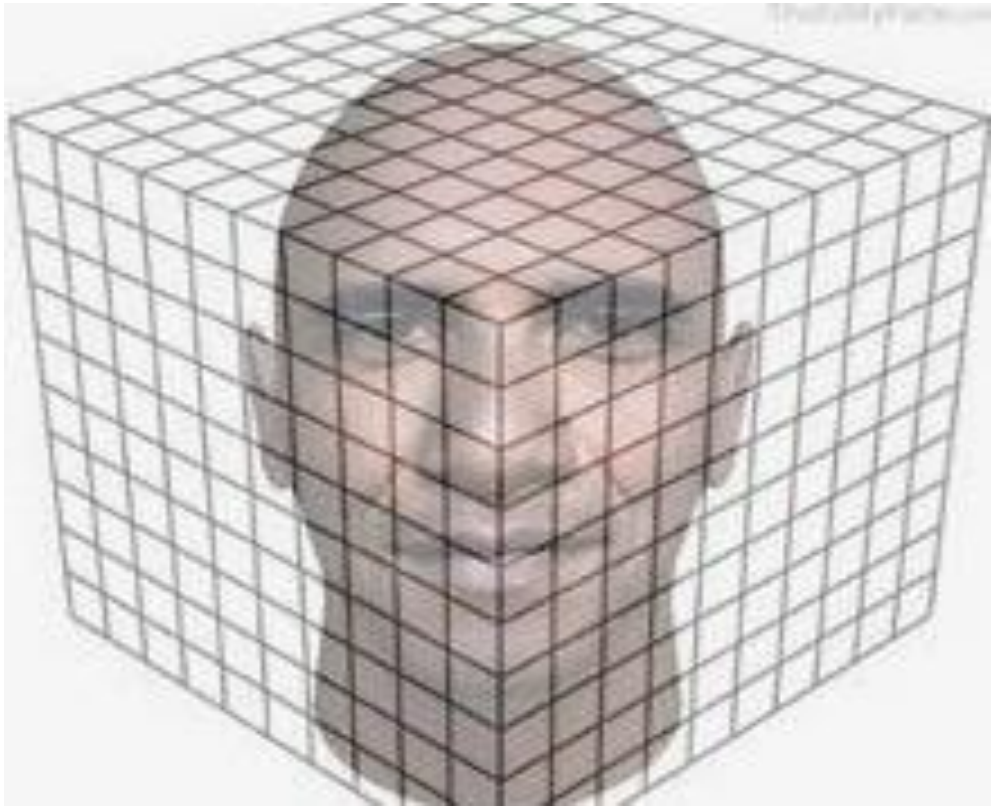
TSDF



d = [pixel depth] – [distance from sensor to voxel]

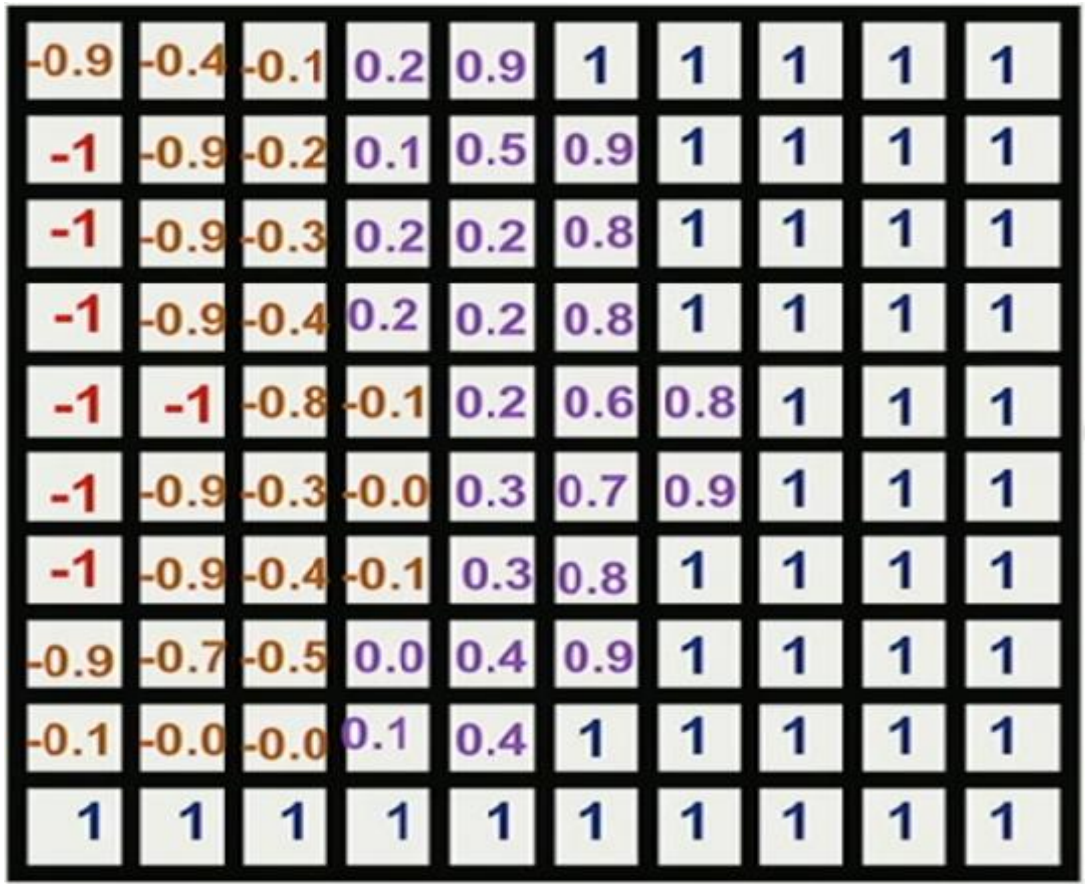
TSDF





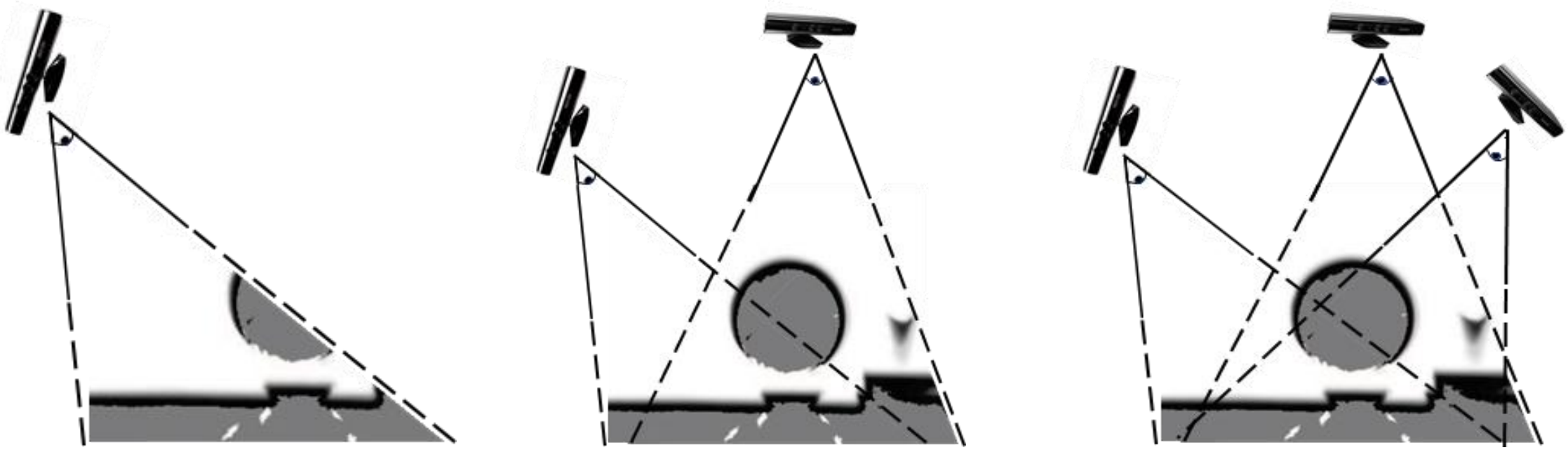
d = [pixel depth] – [distance from sensor to voxel]

TSDF



d = [pixel depth] – [distance from sensor to voxel]

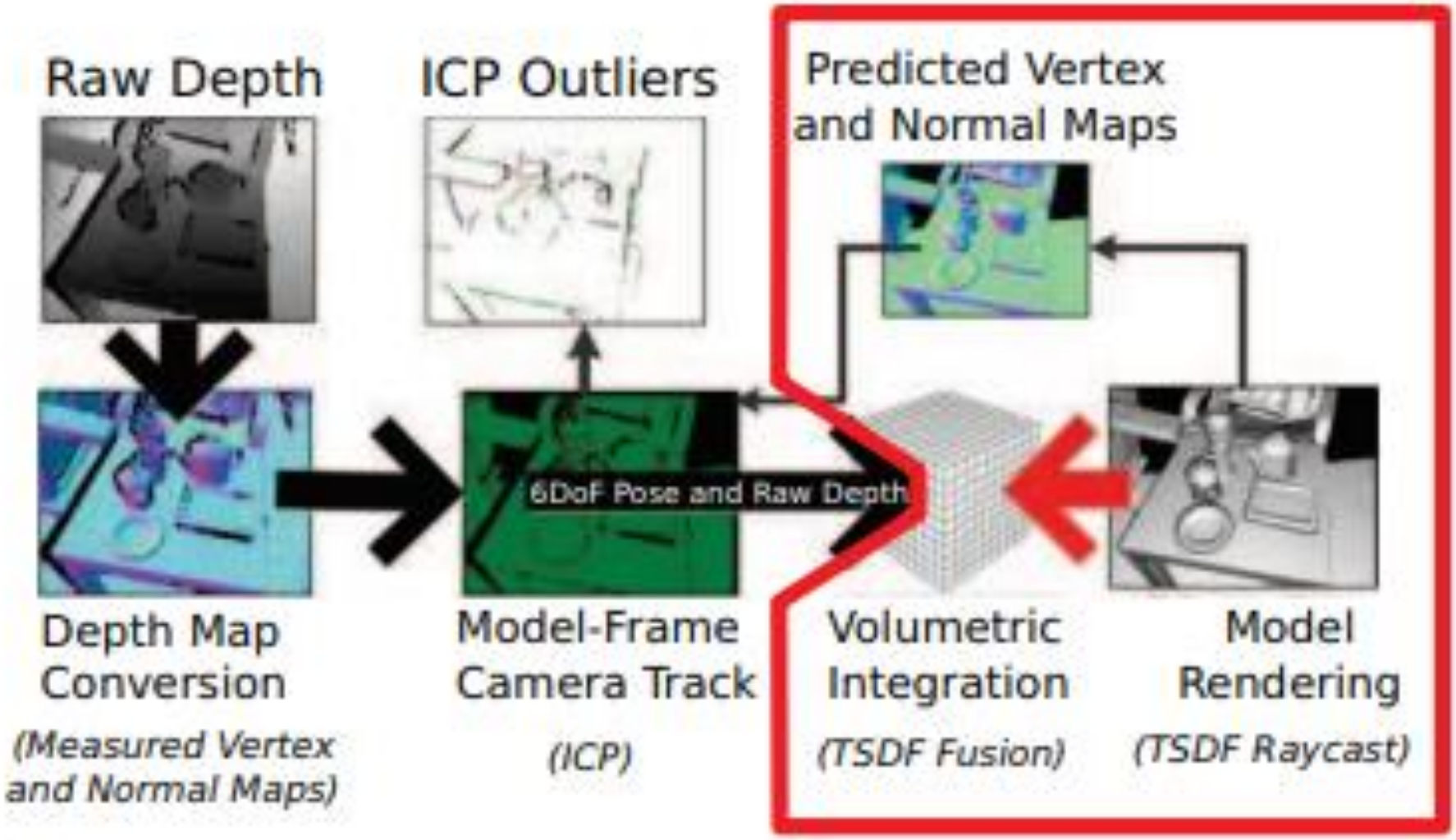
TSDF FUSION



$$F_k(\mathbf{p}) = \frac{W_{k-1}(\mathbf{p})F_{k-1}(\mathbf{p}) + W_{R_k}(\mathbf{p})F_{R_k}(\mathbf{p})}{W_{k-1}(\mathbf{p}) + W_{R_k}(\mathbf{p})}$$

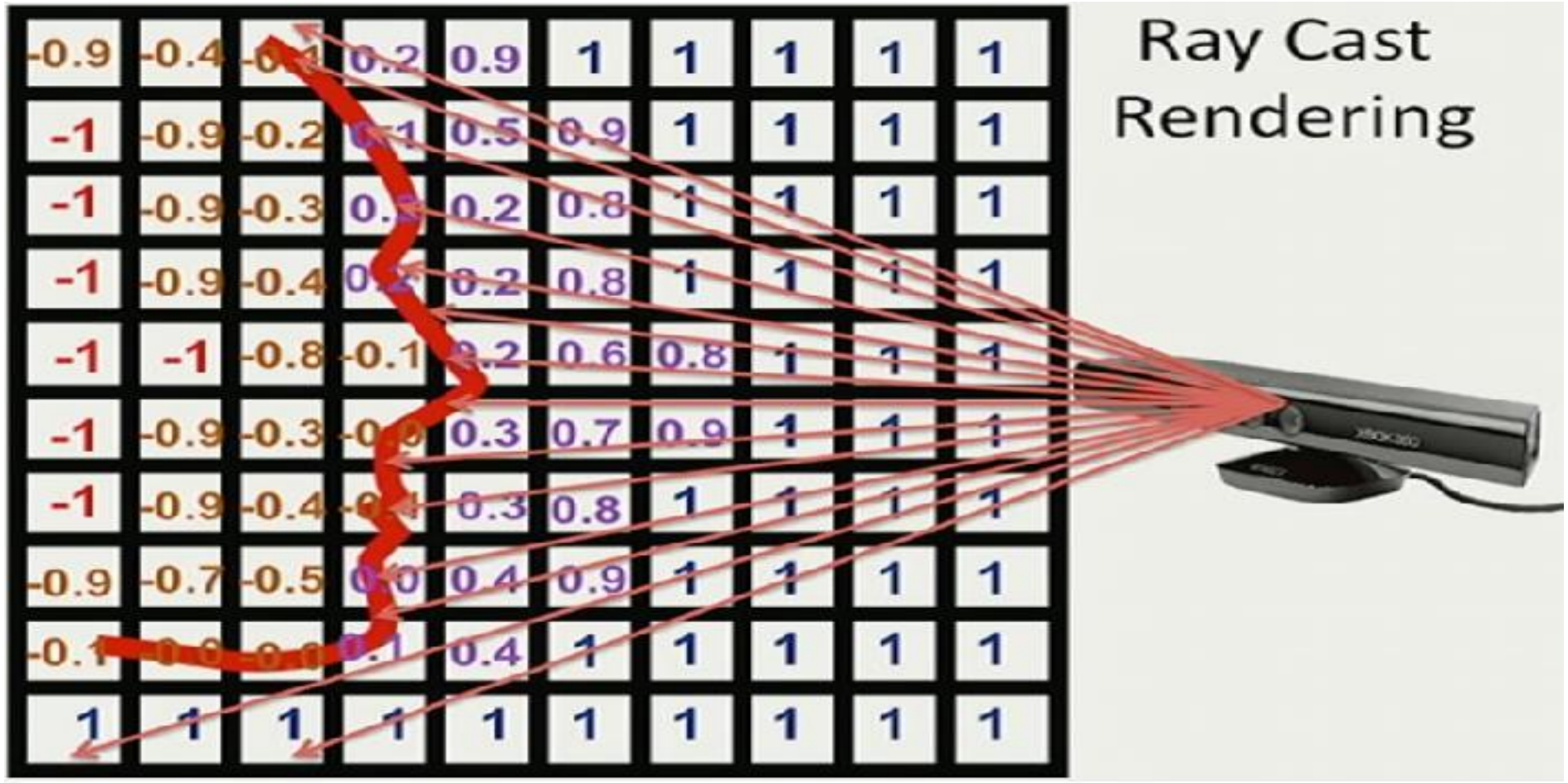
$$W_k(\mathbf{p}) = W_{k-1}(\mathbf{p}) + W_{R_k}(\mathbf{p})$$

RAYCASTING

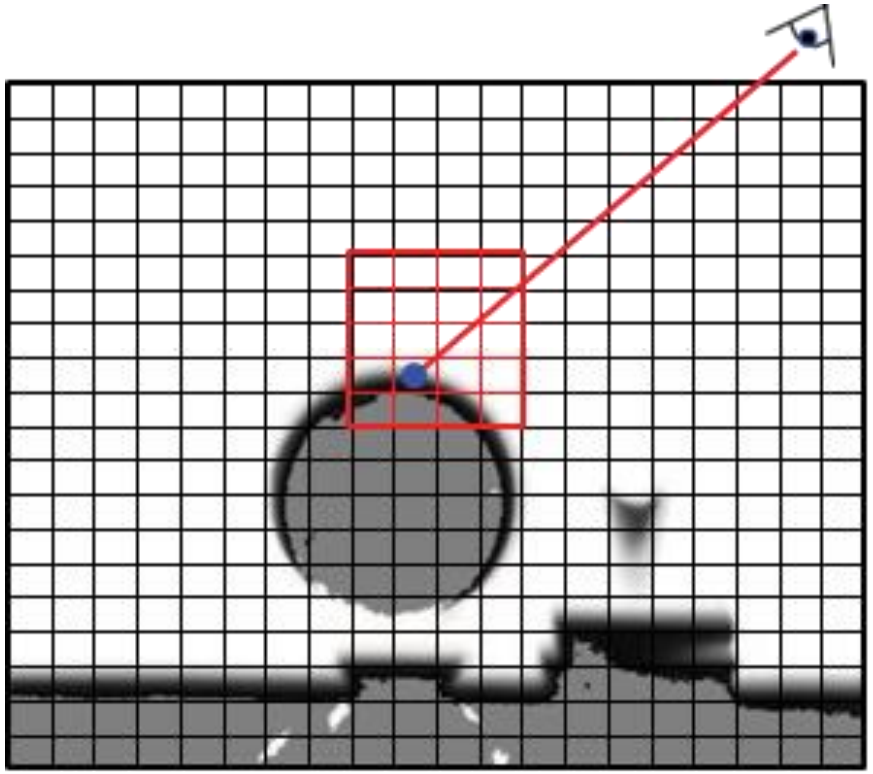

Raw Depth
ICP Outliers
Predicted Vertex and Normal Maps
6DoF Pose and Raw Depth
Depth Map Conversion
(Measured Vertex and Normal Maps)
Model-Frame Camera Track
(ICP)
Volumetric Integration
(TSDF Fusion)
Model Rendering
(TSDF Raycast)

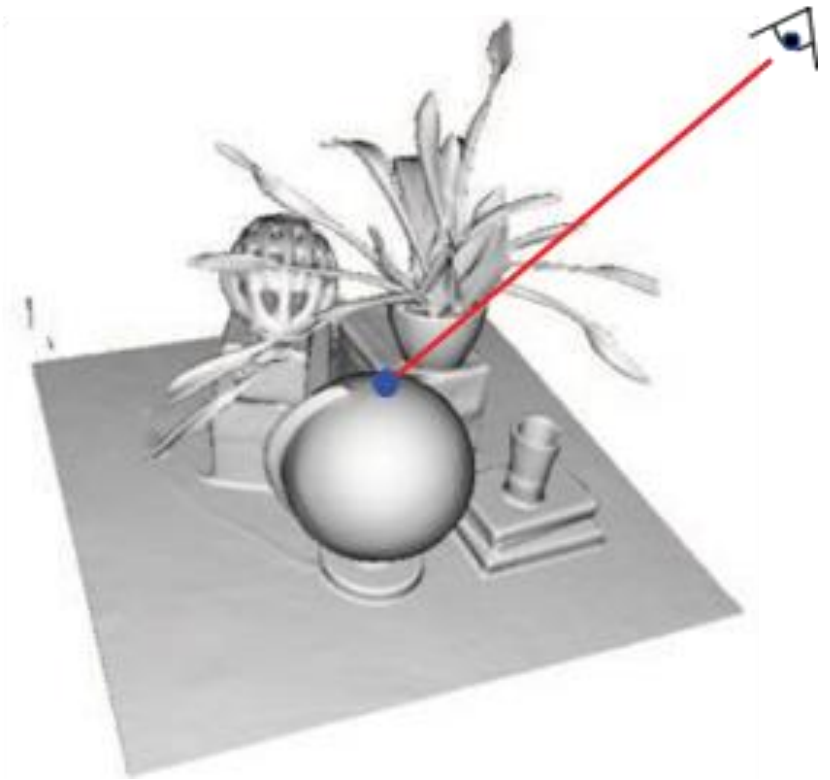
RAYCASTING

- Cast a ray for each pixel of the picture being rendered

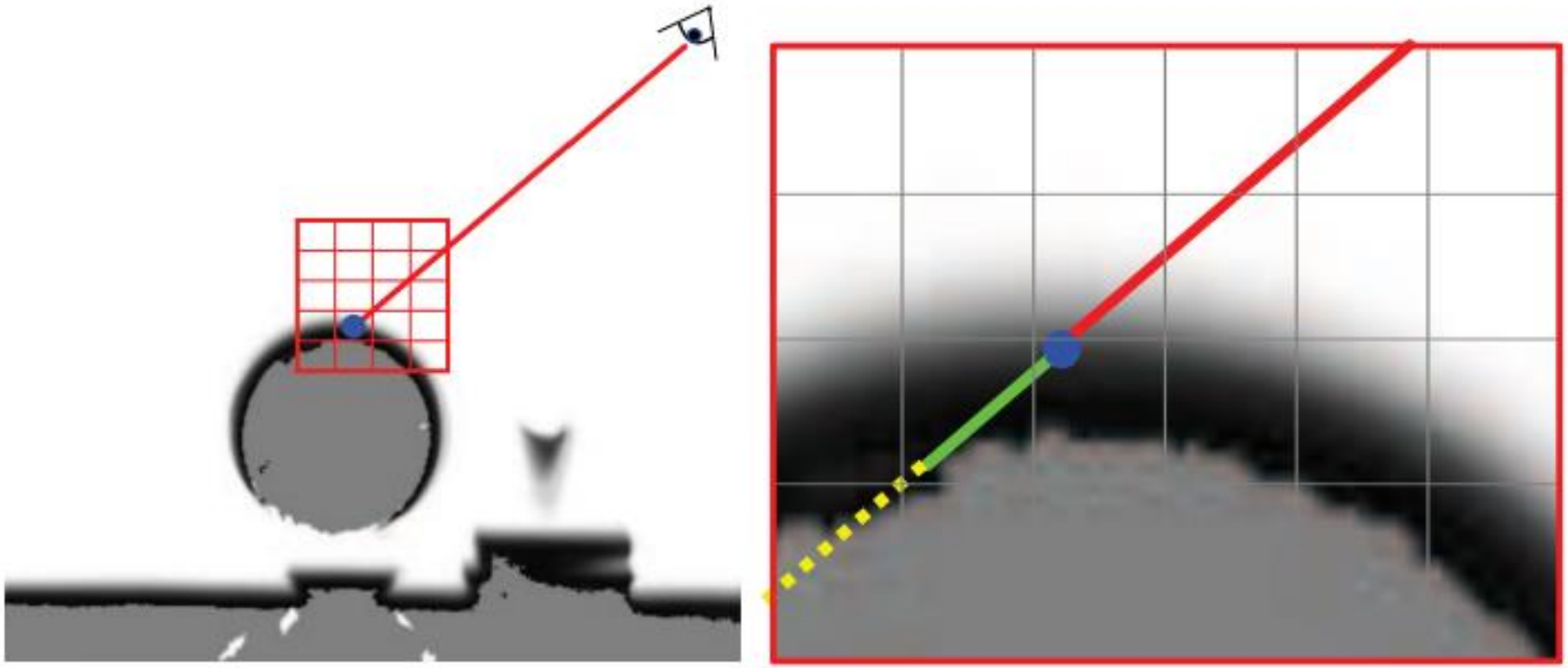


RAYCASTING

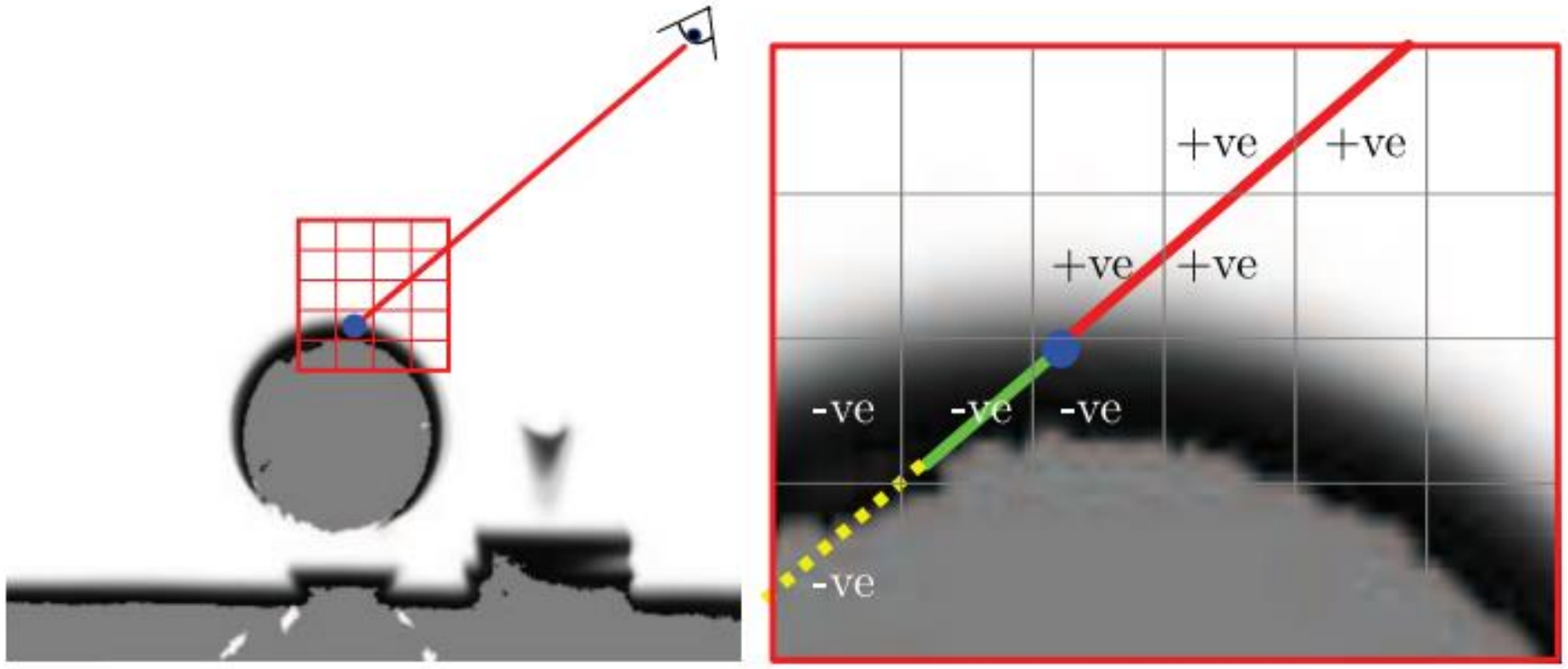




RAYCASTING

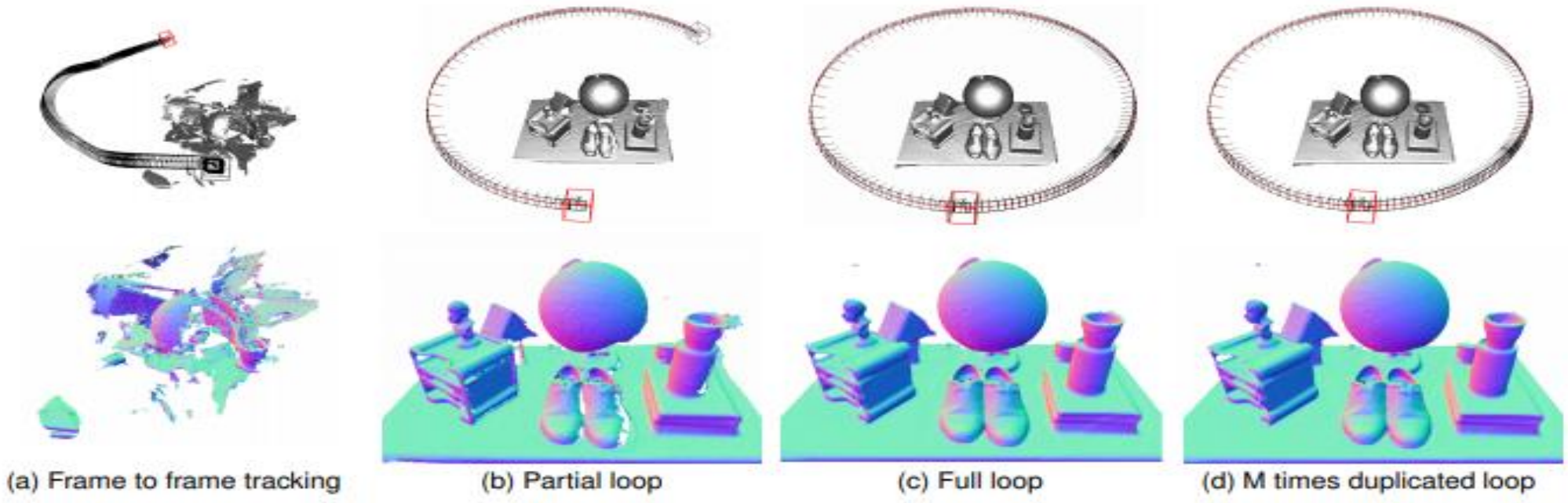


RAYCASTING



DRIFT HANDLING

- Drift would have happened if tracking was done from fame to previous frame
- Tracking is done on built model



EXPERIMENTAL RESULTS

LIMITATIONS - KINECT

- Doesn’t work for large areas
- Doesn’t work for far away objects
- Doesn’t work outdoors
- Struggles with surfaces with little 3D features

LIMITATIONS

- Voxel model isn’t very flexible
- Drift is still possible for long exploratory loops as there is no explicit loop closure.
- Can’t properly model deformations
- Requires a really powerful gamer PC

KINTINUOUS

- Altering KinectFusion which allows region of space being mapped to vary dynamically
- Extract a dense point cloud from region that leave KinectFusion volume
- Add these resulting points to a triangular mesh representation of the environment

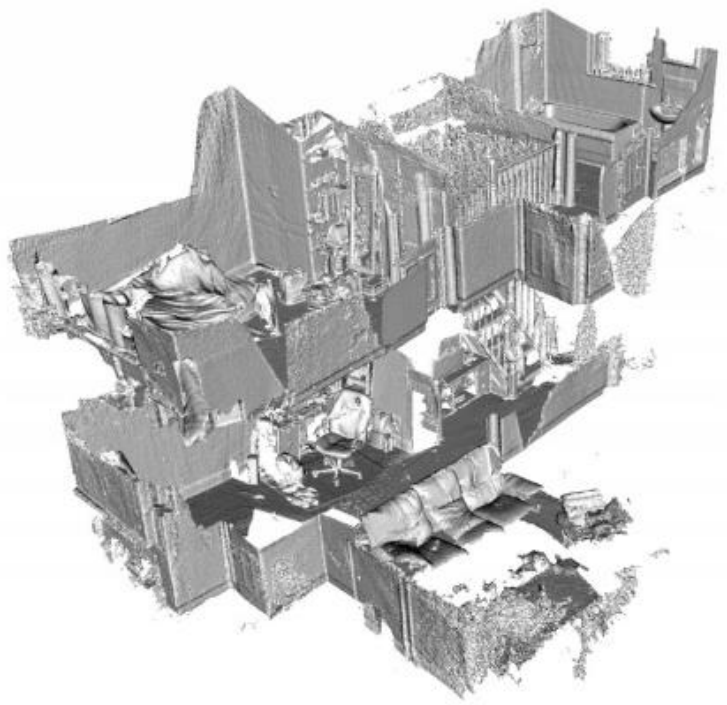
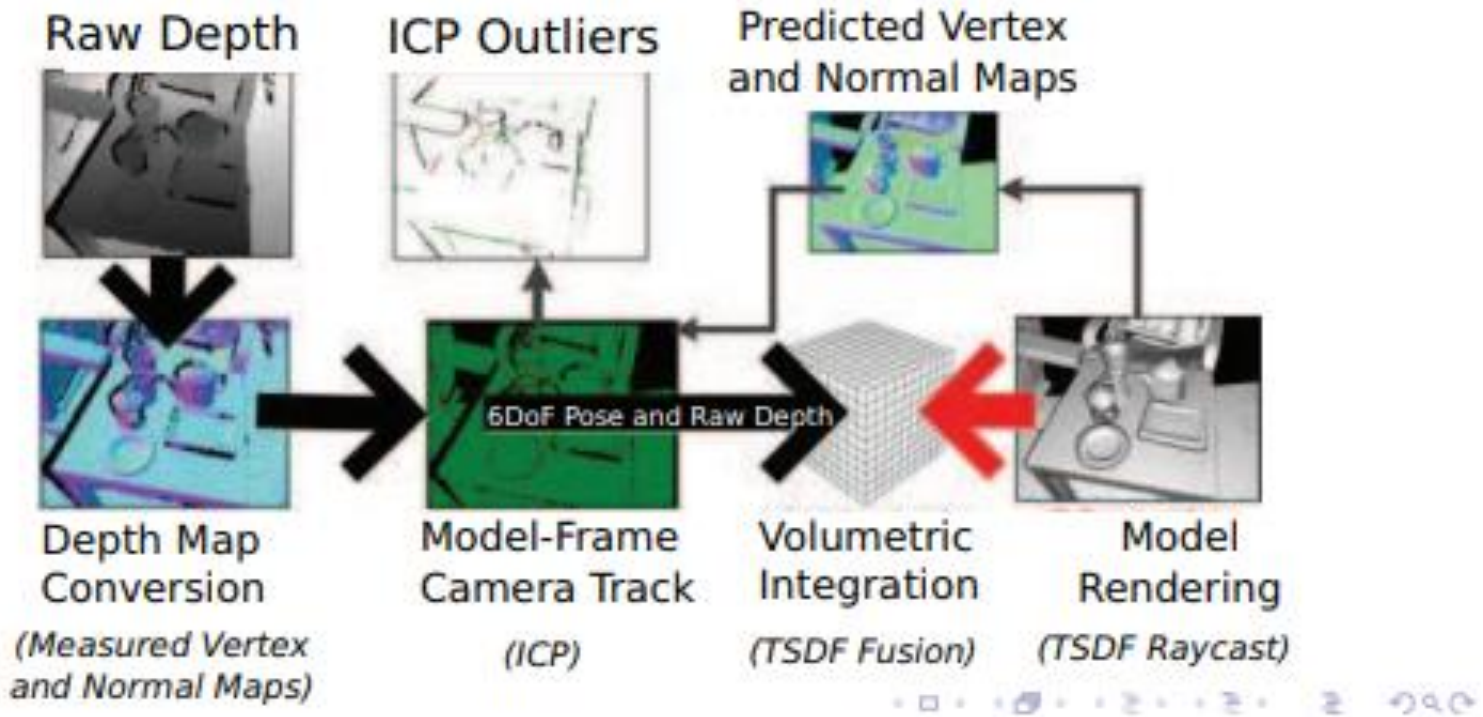


Fig. 1. Real-time 6DOF extended scale map reconstruction of a dataset captured using a handheld Kinect traversing multiple rooms over two floors of an apartment. (see Section V-B)

CONCLUSION

- Real – Time surface reconstruction system
- Speed is achieved through massively parallel GPU implementations
- Augmented Reality + HCI
- Kintinuous improves some limitations

Thank You



- Questions?