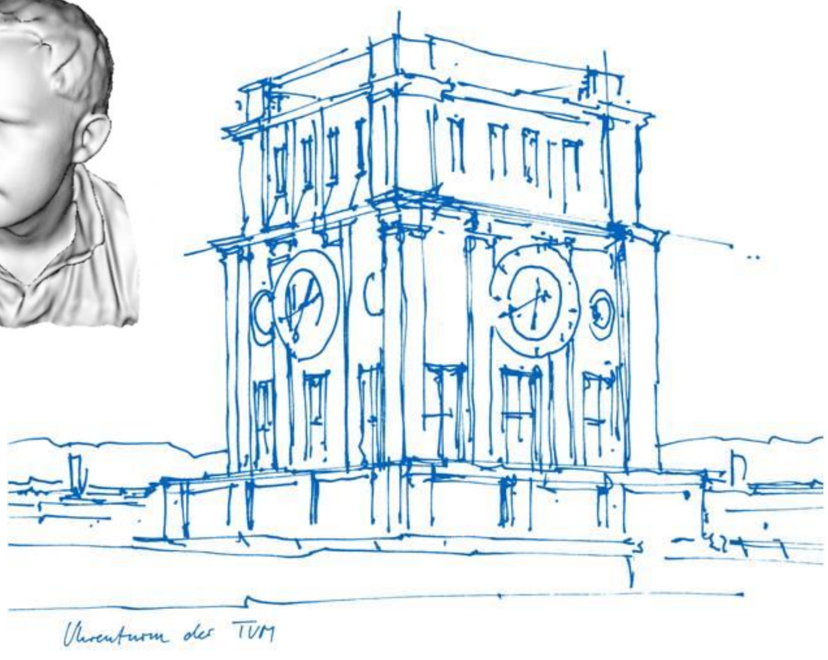
Seminar: Recent Advances in 3D Computer Vision

Wenliang Peng

05.10.2020

KinectFusion: Real-Time Dense Surface Mapping and Tracking



Outline

- Introduction and Motivation

- KinectFusion Algorithm
  - Surface measurement
  - Pose estimation
  - Surface reconstruction update
  - Surface prediction

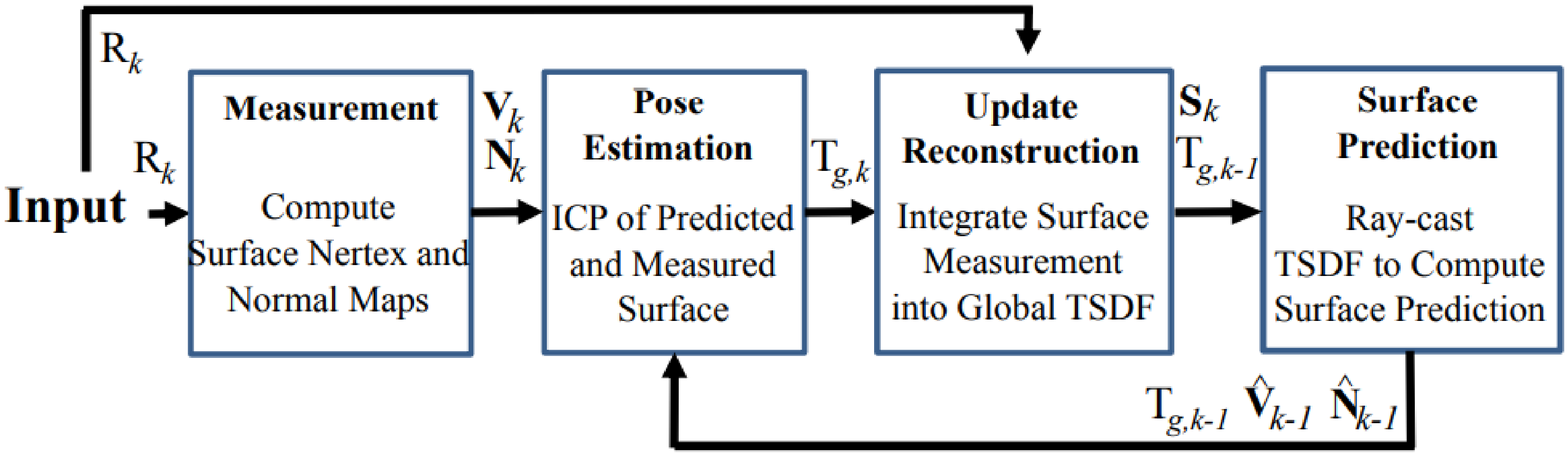
- Experiment and Result

- Conclusion and Outlook

Introduction and Motivation

- SLAM
  - Simultaneous localization and mapping
- Kinect
  - RGB-D Camera
  - depth through either structured light or time of flight
- GPU
  - computation power
  - real time localization and mapping

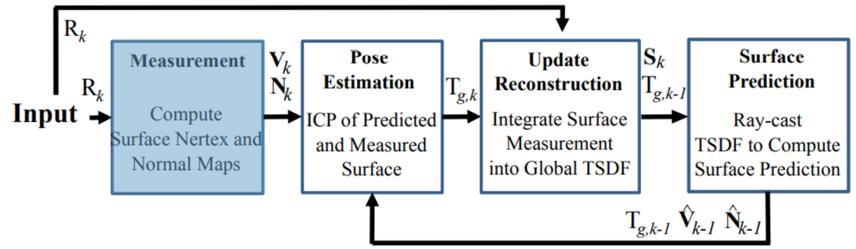
KinectFusion Algorithm Pipeline



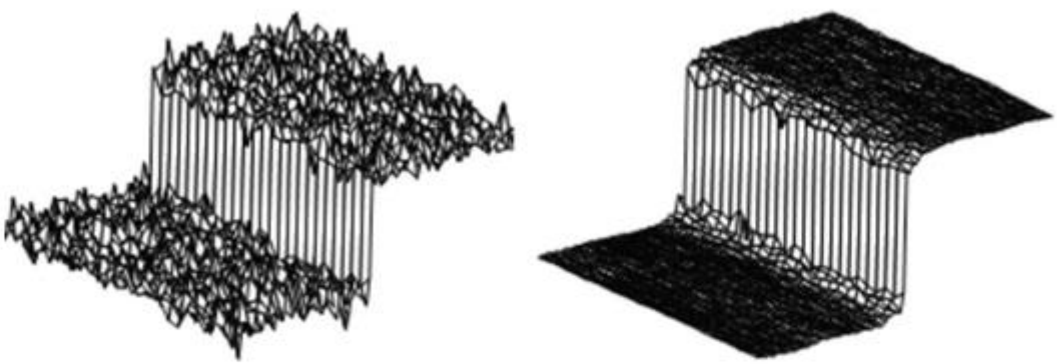
- ICP (Iterative Closest Point), an algorithm for pose estimation
- TSDF (truncated signed distance function), a surface representation based on voxels
- Not Nertex, but Vertex (Misspelled word)

Surface measurement

- Reduction of noise
- Bilateral filter



$$\mathrm{D}_k(\mathbf{u}) = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q}\in\mathscr{U}} \mathscr{N}_{\sigma_s}(\|\mathbf{u}-\mathbf{q}\|_2)\,\mathscr{N}_{\sigma_r}(\|\mathrm{R}_k(\mathbf{u})-\mathrm{R}_k(\mathbf{q})\|_2)\,\mathrm{R}_k(\mathbf{q})$$



$\dot{\mathbf{u}} := (\mathbf{u}^\top|1)^\top$

point $\mathbf{p}_k \in \mathbb{R}^3$ in the camera frame

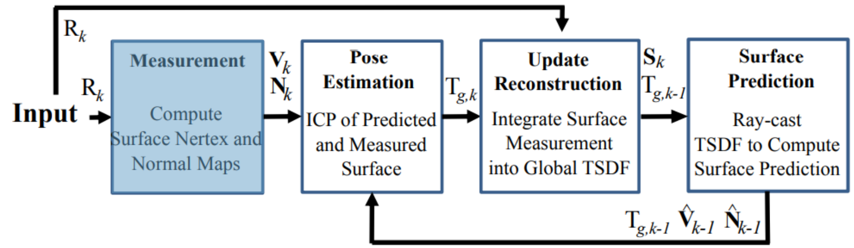
$\mathscr{N}_\sigma(t) = \exp(-t^2\sigma^{-2})$

$W_{\mathbf{p}}$ is a normalizing constant

$\mathbf{q} = \pi(\mathbf{p})$

Surface measurement

- Reduction of noise



- Vertex map V_k

$$\mathbf{V}_k(\mathbf{u}) = \mathrm{D}_k(\mathbf{u})\mathbf{K}^{-1}\dot{\mathbf{u}}$$

$$\dot{\mathbf{u}} := (\mathbf{u}^\top|1)^\top$$

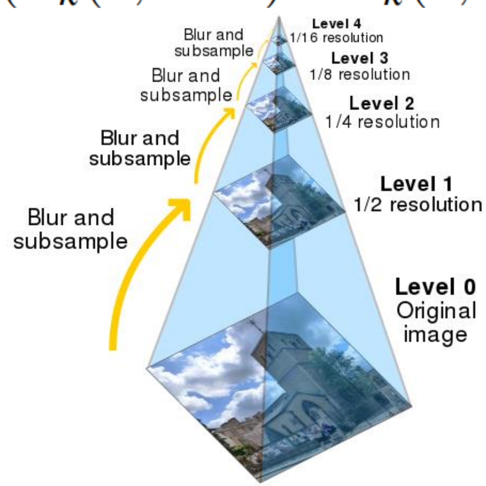
$$v[\mathbf{x}] = \mathbf{x}/\|\mathbf{x}\|_2$$

- Normal map $N_k(u)$

$$\mathbf{N}_k(\mathbf{u}) = v\big[(\mathbf{V}_k(u+1,v) - \mathbf{V}_k(u,v)) \times (\mathbf{V}_k(u,v+1) - \mathbf{V}_k(u,v))\big]$$

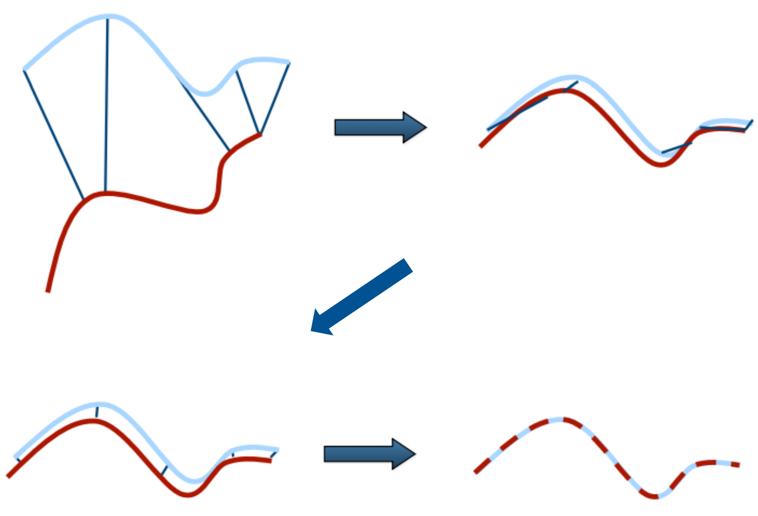
- vertex and normal map pyramid
  - L = 3 level multi-scale representation

$$\mathbf{V}^{l\in[1\ldots L]}, \mathbf{N}^{l\in[1\ldots L]}$$

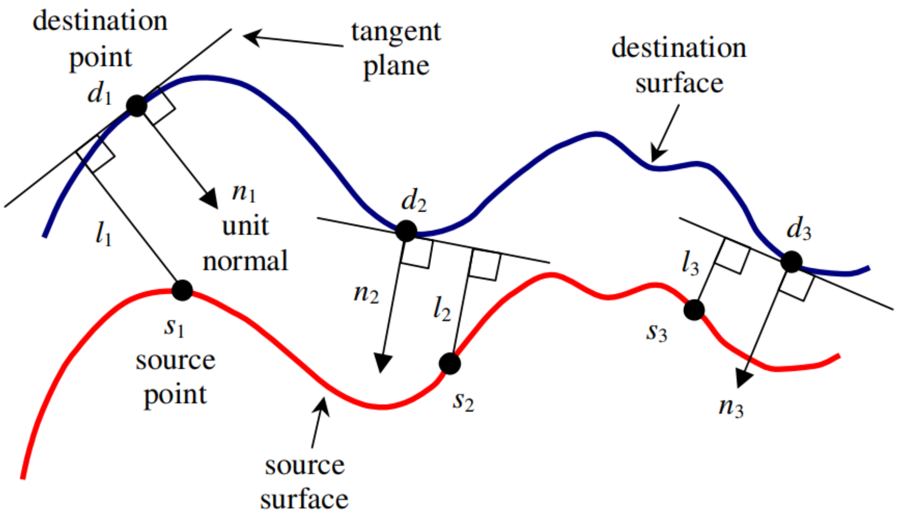


Pose estimation

- ICP (Iterative Closest Point)
- the global point-plane ICP



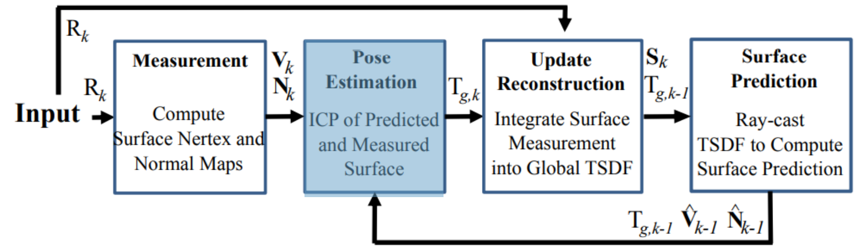
$$\mathbf{M}_{opt} = \arg\min_{\mathbf{M}} \sum_i ((\mathbf{M} \cdot \mathbf{s}_i - \mathbf{d}_i) \bullet \mathbf{n}_i)^2$$



Pose estimation

- the global point-plane ICP

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



$$\hat{\mathbf{u}} = \pi(\mathrm{K}\widetilde{\mathrm{T}}_{k-1,k}\dot{\mathbf{V}}_k(\mathbf{u}))$$

$$\widetilde{\mathrm{T}}^z_{k-1,k} = \mathrm{T}^{-1}_{g,k-1}\widetilde{\mathrm{T}}^z_{g,k}$$

$$\mathbf{M}_{\mathrm{opt}} = \arg\min_{\mathbf{M}} \sum_i ((\mathbf{M} \cdot \mathbf{s}_i - \mathbf{d}_i) \bullet \mathbf{n}_i)^2$$

$$\mathbf{E}(\mathrm{T}_{g,k}) = \sum_{\mathbf{u} \in \mathscr{U}} \left\| \left(\mathrm{T}_{g,k}\dot{\mathbf{V}}_k(\mathbf{u}) - \hat{\mathbf{V}}^g_{k-1}(\hat{\mathbf{u}}) \right)^\top \hat{\mathbf{N}}^g_{k-1}(\hat{\mathbf{u}}) \right\|_2$$

Pose estimation

- The iterative solution of ICP

$$\mathbf{E}(\mathrm{T}_{g,k}) = \sum_{\mathbf{u}\in\mathscr{U}} \left\| \left(\mathrm{T}_{g,k}\dot{\mathbf{V}}_k(\mathbf{u}) - \hat{\mathbf{V}}^g_{k-1}(\hat{\mathbf{u}}) \right)^\top \hat{\mathbf{N}}^g_{k-1}(\hat{\mathbf{u}}) \right\|_2$$

$$\mathrm{T}_{g,k} = \begin{bmatrix} \mathrm{R}_{g,k} & \mathbf{t}_{g,k} \\ \mathbf{0}^\top & 1 \end{bmatrix} \in \mathbb{SE}_3 \quad \Rightarrow \quad \widetilde{\mathrm{T}}^z_{g,k} = \widetilde{\mathrm{T}}^z_{inc}\widetilde{\mathrm{T}}^{z-1}_{g,k}$$

$$\widetilde{\mathrm{T}}^z_{inc} = \left[\widetilde{\mathrm{R}}^z \,|\, \tilde{\mathbf{t}}^z\right] = \begin{bmatrix} 1 & \alpha & -\gamma & t_x \\ -\alpha & 1 & \beta & t_y \\ \gamma & -\beta & 1 & t_z \end{bmatrix}$$

Small-angle approximation

$$\widetilde{\mathbf{V}}^g_k(\mathbf{u}) = \widetilde{\mathrm{T}}^{z-1}_{g,k}\dot{\mathbf{V}}_k(\mathbf{u})$$

$$\widetilde{\mathrm{T}}^z_{g,k}\dot{\mathbf{V}}_k(\mathbf{u}) = \widetilde{\mathrm{R}}^z\widetilde{\mathbf{V}}^g_k(\mathbf{u}) + \tilde{\mathbf{t}}^z = \mathbf{G}(\mathbf{u})\mathbf{x} + \widetilde{\mathbf{V}}^g_k(\mathbf{u})$$

$$\mathbf{x} = (\beta,\gamma,\alpha,t_x,t_y,t_z)^\top \in \mathbb{R}^6 \qquad \mathbf{G}(\mathbf{u}) = \left[\left[\widetilde{\mathbf{V}}^g_k(\mathbf{u})\right]_\times \,\middle|\, \mathbf{I}_{3\times 3}\right]$$

Pose estimation

- The iterative solution of ICP

$$\mathbf{E}(\mathrm{T}_{g,k}) = \sum_{\mathbf{u}\in\mathscr{U}} \left\| \left(\mathrm{T}_{g,k}\dot{\mathbf{V}}_k(\mathbf{u}) - \hat{\mathbf{V}}^g_{k-1}(\hat{\mathbf{u}})\right)^\top \hat{\mathbf{N}}^g_{k-1}(\hat{\mathbf{u}}) \right\|_2 \qquad \min_{\mathbf{x}\in\mathbb{R}^6} \sum_{\Omega_k(\mathbf{u})\neq\text{null}} \|E\|_2^2$$

$$E = \hat{\mathbf{N}}^g_{k-1}(\hat{\mathbf{u}})^\top \left(\mathbf{G}(\mathbf{u})\mathbf{x} + \widetilde{\mathbf{V}}^g_k(\mathbf{u}) - \hat{\mathbf{V}}^g_{k-1}(\hat{\mathbf{u}})\right)$$

First-order optimality condition

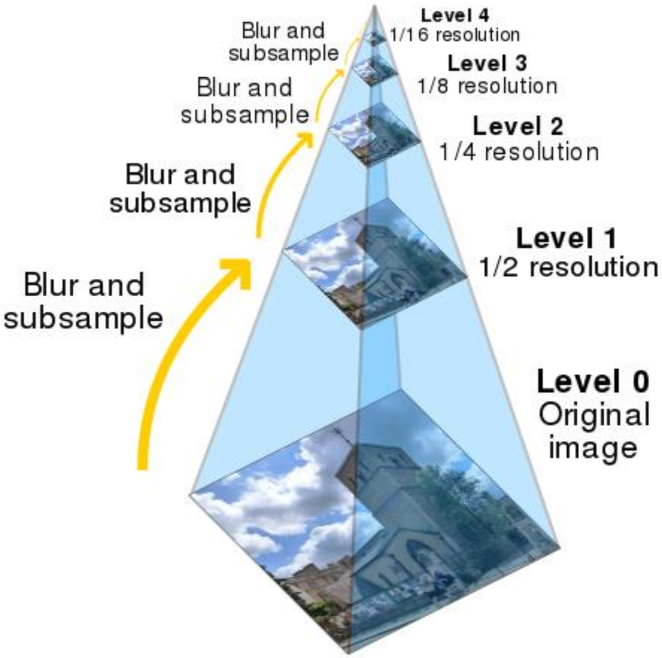
$$\sum \left(\mathbf{A}^\top\mathbf{A}\right)\mathbf{x} = \sum \mathbf{A}^\top\mathbf{b} \qquad \begin{aligned} \mathbf{A}^\top &= \mathbf{G}^\top(\mathbf{u})\hat{\mathbf{N}}^g_{k-1}(\hat{\mathbf{u}}), \\ \mathbf{b} &= \hat{\mathbf{N}}^g_{k-1}(\hat{\mathbf{u}})^\top\left(\hat{\mathbf{V}}^g_{k-1}(\hat{\mathbf{u}}) - \widetilde{\mathbf{V}}^g_k(\mathbf{u})\right) \end{aligned}$$

$$\mathbf{x} = (\beta, \gamma, \alpha, t_x, t_y, t_z)^\top \in \mathbb{R}^6$$

$$z_{max} = [4, 5, 10] \quad \text{levels } [3, 2, 1]$$

Pose estimation

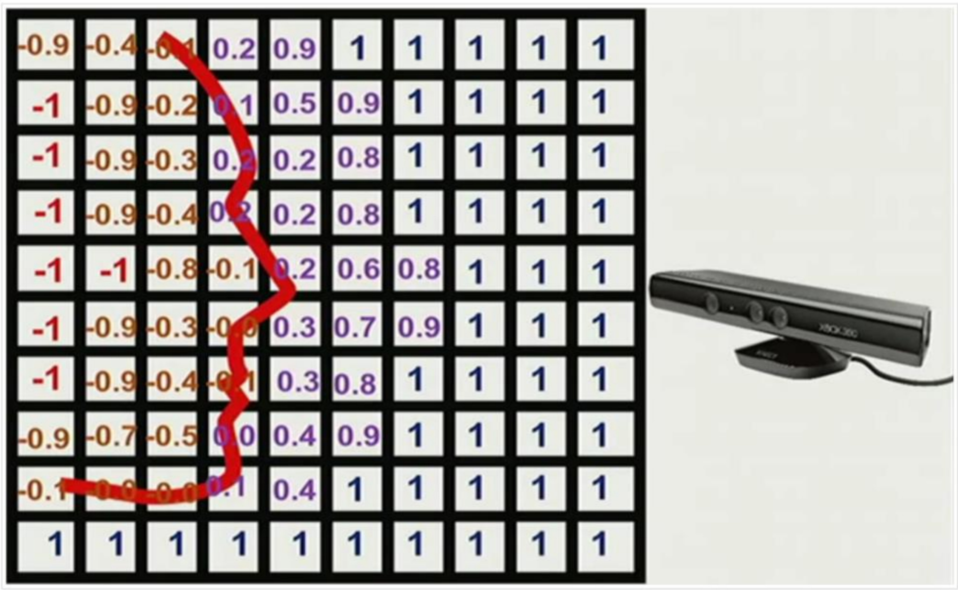
- Image pyramid
  - problem is highly non-convex
  - algorithm might be trapped in a bad local minimum
  - a good initialization is needed for the optimization
  - coarse to fine scheme

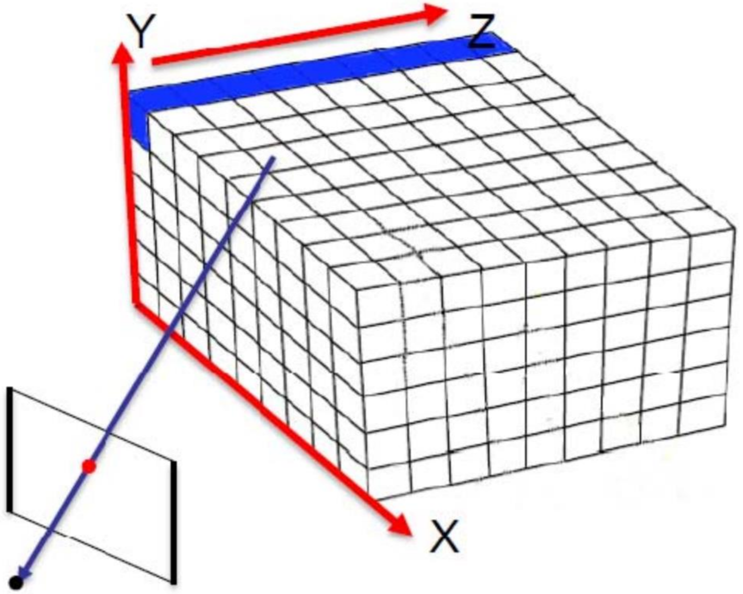


Surface reconstruction update

- TSDF (truncated signed distance function)

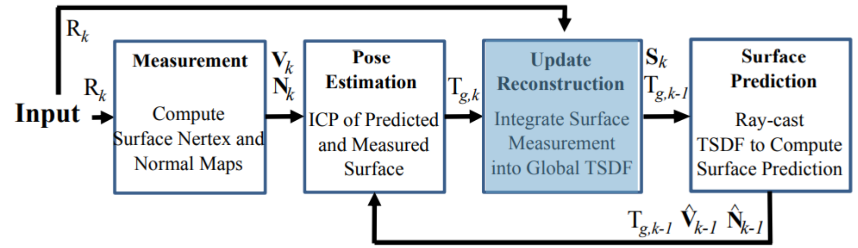
$$\mathbf{S}_k(\mathbf{p}) \mapsto [\mathrm{F}_k(\mathbf{p}), \mathrm{W}_k(\mathbf{p})]$$



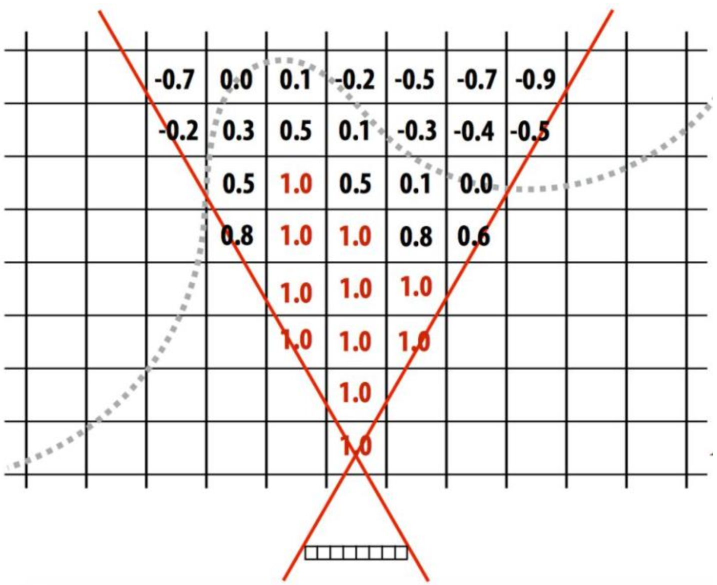


Surface reconstruction update

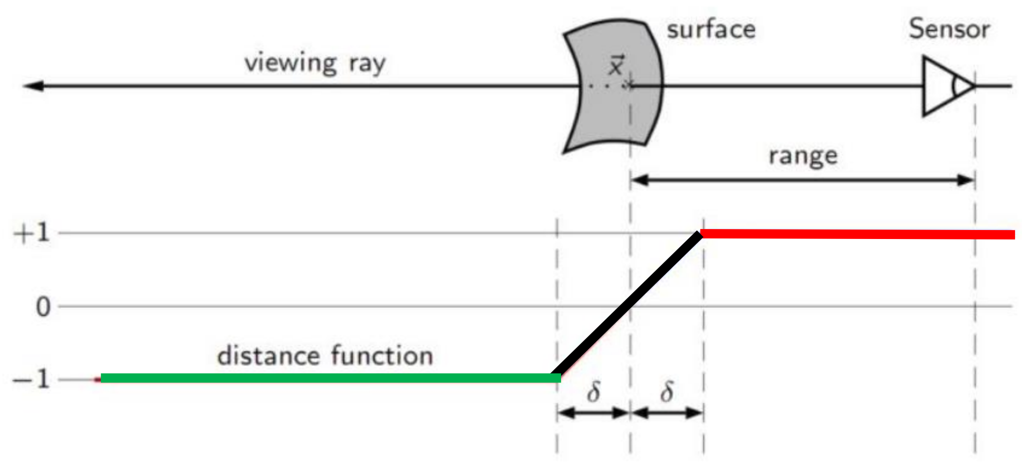
- The point p in the global frame g



$$\mathbf{S}_k(\mathbf{p}) \mapsto [\mathrm{F}_k(\mathbf{p}), \mathrm{W}_k(\mathbf{p})]$$



$$
\begin{aligned}
\mathrm{F}_{\mathrm{R}_k}(\mathbf{p}) &= \Psi\left(\lambda^{-1}\|(\mathbf{t}_{g,k}-\mathbf{p})\|_2 - \mathrm{R}_k(\mathbf{x})\right) ,\\
\lambda &= \|\mathrm{K}^{-1}\dot{\mathbf{x}}\|_2 ,\\
\mathbf{x} &= \left\lfloor \pi\left(\mathrm{K}\mathrm{T}_{g,k}^{-1}\mathbf{p}\right)\right\rfloor \text{ nearest neighbour lookup } \lfloor . \rfloor \\
\Psi(\eta) &= \begin{cases} \min\left(1, \frac{\eta}{\mu}\right)\mathrm{sgn}(\eta) & \text{iff } \eta \geq -\mu \\ null & otherwise \end{cases}
\end{aligned}
$$



Surface reconstruction update

- The global fusion of all depth maps in the volume

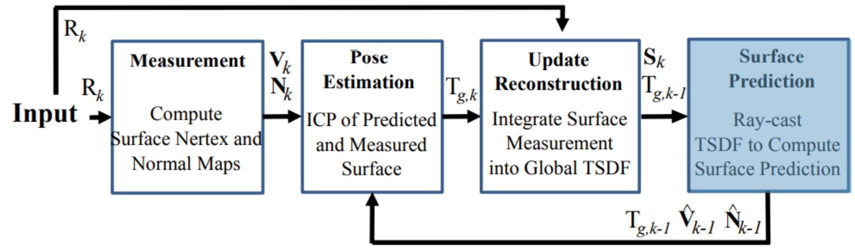
$$\min_{F \in \mathscr{F}} \sum_{k} \| W_{R_k} F_{R_k} - F) \|_2$$

- real-time sensor tracking and surface reconstruction

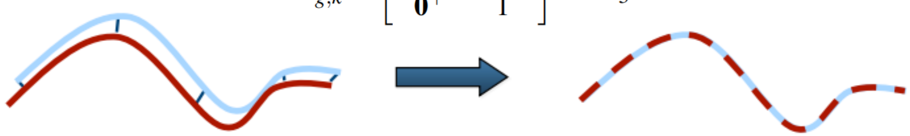
$$\begin{aligned} F_k(\mathbf{p}) &= \frac{W_{k-1}(\mathbf{p}) F_{k-1}(\mathbf{p}) + W_{R_k}(\mathbf{p}) F_{R_k}(\mathbf{p})}{W_{k-1}(\mathbf{p}) + W_{R_k}(\mathbf{p})} \\ W_k(\mathbf{p}) &= W_{k-1}(\mathbf{p}) + W_{R_k}(\mathbf{p}) \end{aligned}$$

Surface Prediction

- vertex map $\hat{V}_{k-1}^{g}$
  - Each pixel's corresponding ray
  - starting from the minimum depth
  - stopping when a zero crossing
  - e.g. blue one is the new vertex map, red one is ray-casting of the global model



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

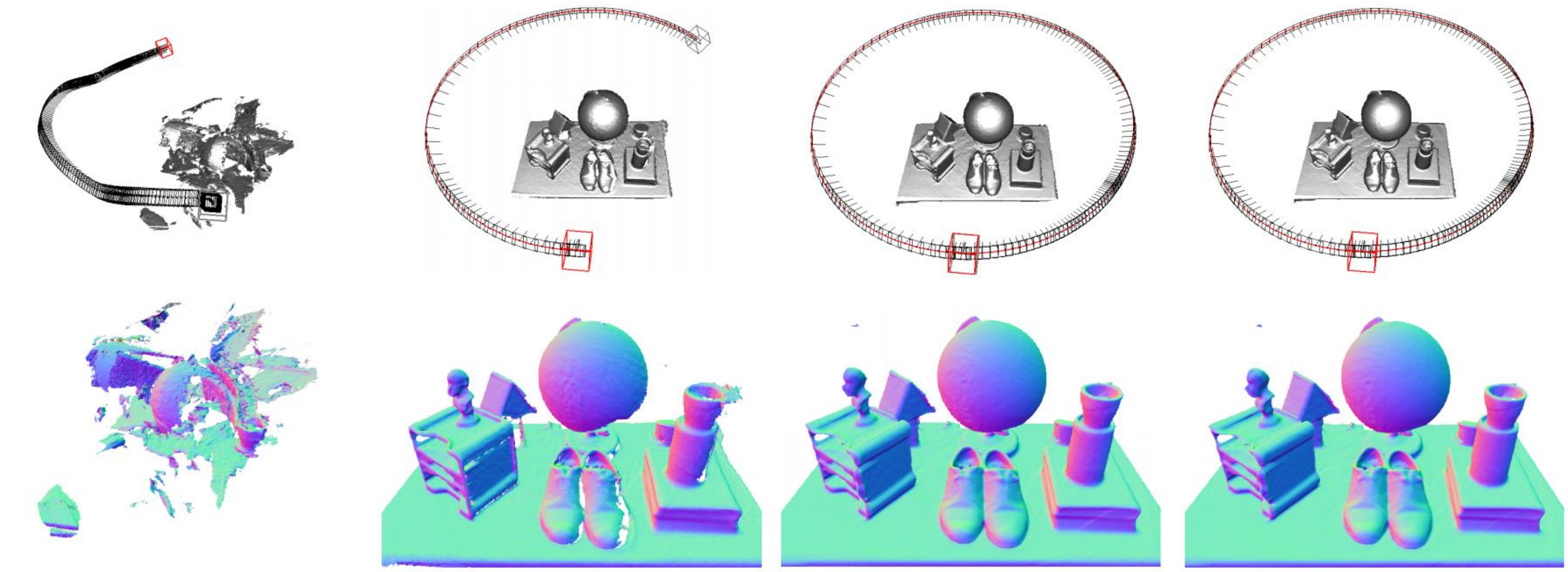


- Normal vectors $\hat{N}_{k-1}^{g}$

$$\mathrm{R}_{g,k}\hat{\mathbf{N}}_k = \hat{\mathbf{N}}_k^g(\mathbf{u}) = \nu\left[\nabla \mathrm{F}(\mathbf{p})\right], \ \nabla \mathrm{F} = \left[\frac{\partial \mathrm{F}}{\partial x}, \frac{\partial \mathrm{F}}{\partial y}, \frac{\partial \mathrm{F}}{\partial z}\right]^\top$$

Experiment and Result

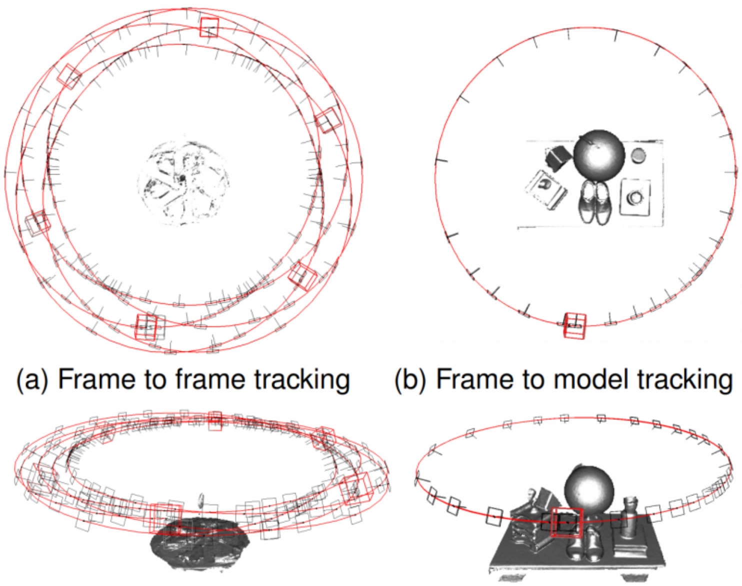
- Experiment setup
  - N = 560 frames over ≈ 19 seconds, Kinect sensor is fixed, turntable
  - reconstruction resolution of 256^3 voxels
- Experiment



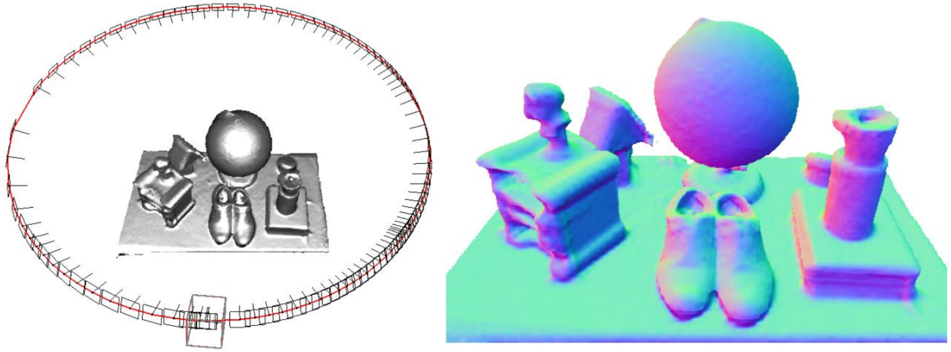
(a) Frame to frame tracking (b) Partial loop (c) Full loop (d) M times duplicated loop

Experiment and Result

- Experiment



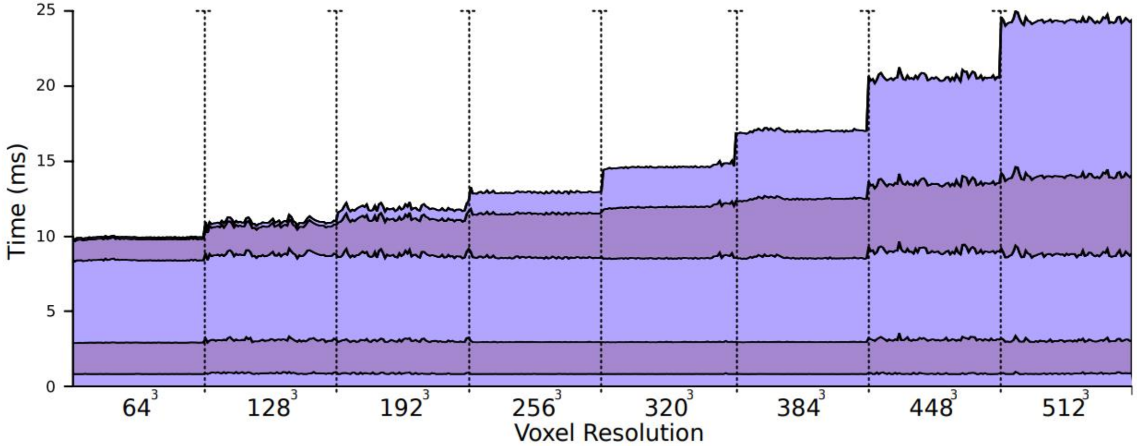
every 8th frame



every 6th sensor frame, 64^3 voxels

Experiment and Result

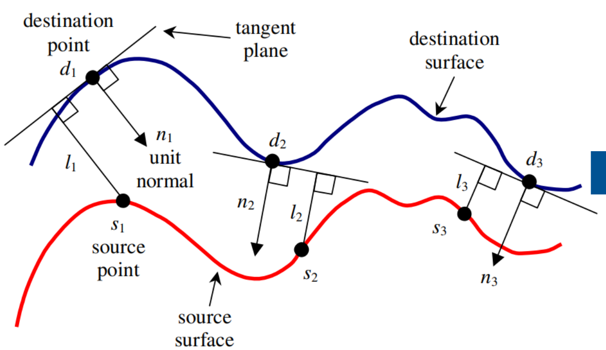
- Experiment

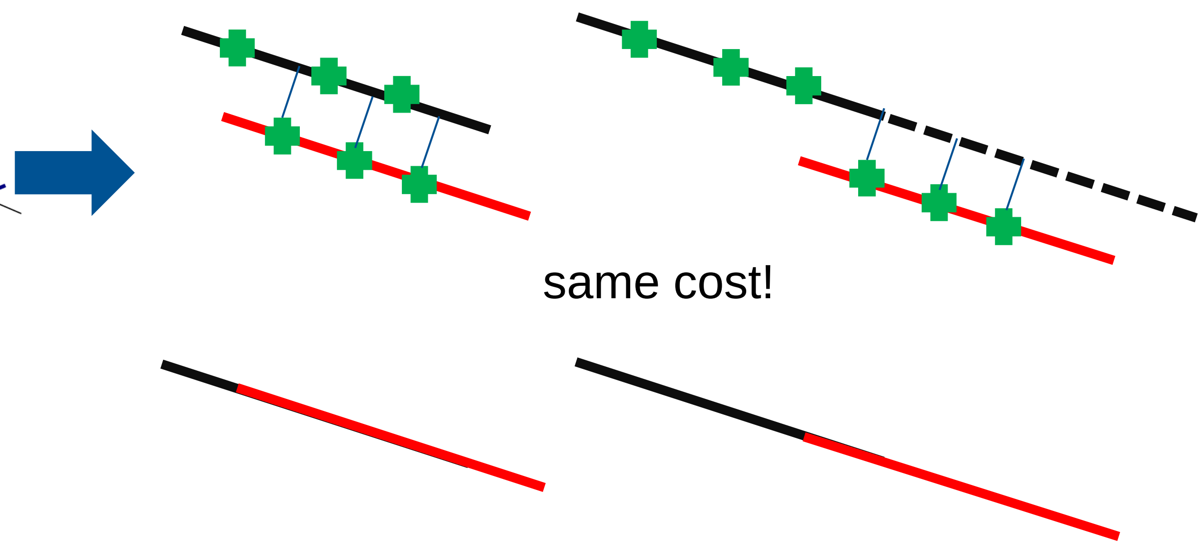


- 3 cubic meters, cumulative timing results (from bottom to top) :
  - pre-processing raw data
  - multi-scale data-associations
  - multi-scale pose optimizations
  - ray-casting the surface prediction
  - surface measurement integration

Experiment and Result

- Failure
  - the sensor is faced by a large planar scene
  - the translation is free to take any vector in the direction of the plane, and it will not change the cost
  - infinitely many possible minimum for the cost





same cost!

- Possible solution
  - add photometric cost

different translation!

Conclusion and Outlook

- Conclusion
  - up-to-date surface representation fusing all registered data from previous scans.
  - accurate and robust tracking of the camera pose by aligning all depth points with the complete scene model.
  - parallel algorithms for both tracking and mapping, taking full advantage of commodity GPU processing hardware.

- Outlook
  - reconstruction of largescale models
  - automatic semantic segmentation

Thank you！
Any question?