Progress Report

Sun Qinxuan

March 3, 2017

To be addressed

- Plane Feature based SLAM
  - Plane extraction (geodesic dome)
  - Plane matching (interpretation tree)
  - Camera pose estimation (shadows on plane)
  - Loop closing and map optimization (?)

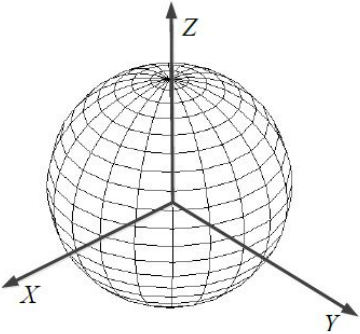
Plane extraction

Plane extraction

- orientation histogram

$$\begin{cases} \theta = \arccos(n_z) \\ \varphi = \operatorname{atan2}(n_y, n_x) \end{cases}$$

- RANSAC to fit plane.



Plane extraction

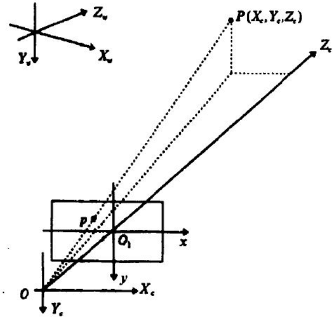
Surface normal distribution

- $\forall P(X_c, Y_c, Z_c)$, the direction vector from P to O_c is

$$\mathbf{d}_{PO} = -\frac{1}{\eta}\left[(u-u_0)p_x, (v-v_0)p_y, f\right]^T$$

  where (u, v) is the pixel coordinate, and η is the normalization factor.
- $\mathbf{n}_P$ is the surface normal estimated at $\mathbf{d}_{PO}$.
- $\mathbf{n}_P$ satisfies

$$\mathbf{n}_P^T \mathbf{d}_{PO} > 0.$$



Plane extraction

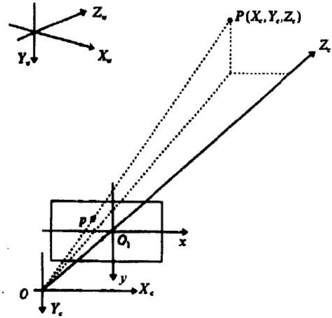
Surface normal distribution

- rewrite it as

$$(u-u_0)p_x n_x + (v-v_0)p_y n_y + f n_z < 0$$

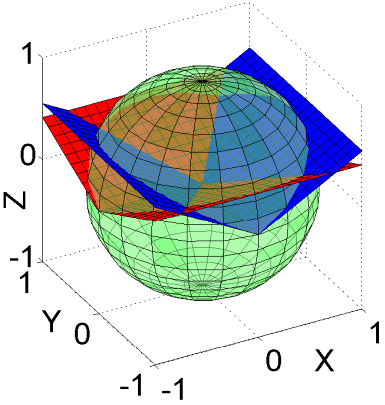
holds true for each (u,v) on image plane.

-



Plane extraction

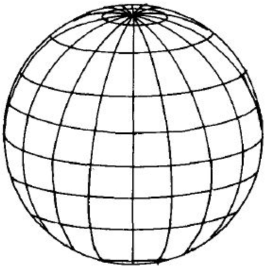
Surface normal distribution



Plane extraction

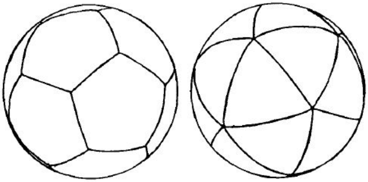
Tesselation of a Sphere

- Ideally the cells should satisfy the following criteria:
  1) All cells have the same shape and area.
  2) The division should be fine enough to provide good angular resolution.
  3) For some rotations, the cells should be brought into coincidence with themselves.



Plane extraction
Tesselation of a Sphere

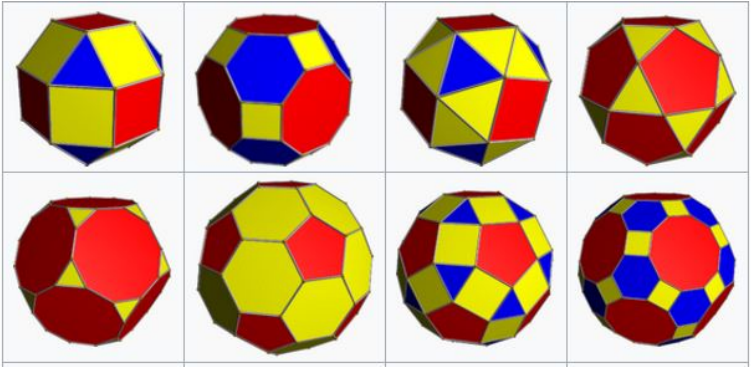
- Tesselations based on regular polyhedra:
  - only five regular solids: tetrahedron, hexahedron, octahedron, dodecahedron, and icosahedron.
  - only 13 semi-regular polyhedra: five truncated regular polyhedra, cuboctahedron, icosidodecahedron, snub cuboctahedron, snub icosidodecahedron, truncated cuboctahedron, rhombicuboctahedron, truncated icosidodecahedron, and the rhombicosidodecahedron.
  - They do not provide us with fine enough tesselations.



(a) (b)

Plane extraction

Tesselation of a Sphere



Plane extraction
Tesselation of a Sphere

- Geodesic dome.
  - Start from polyhedra composed of triangular facets. (isosahedron, pentakis dodecahedron)
  - Each of the edges of the triangular cells of the original polyhedron are divided into f sections. (f is called the frequency of the geodesic division.)
  - The result is that each face is divided into f^2 triangles.
  - It is possible to proceed hierarchically, particularly if the frequency is a power of two.

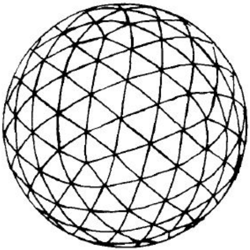
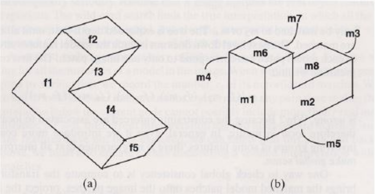


Fig. 18. Tesselation of the Gaussian sphere using a frequency-four geodesic tesselation based on the isosahedron. (There are $16 \times 20 = 320$ faces.)

Plane matching

Interpretation tree

- Problem statement
  - Given
    - A list of plane feature descriptors from the reference scan.
    - A list of plane feature descriptors form the current scan.
    - A list of geometric constraints that the plane features must satisfy.
  - Find a mapping between plane features from two scans such that the features match correctly and satisfy the geometric constraints.



Plane matching

Interpretation tree

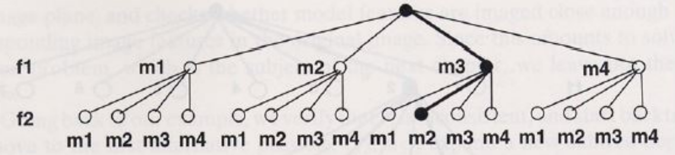


Figure 10.1 An interpretation tree representing the complete search space for a problem involving two image features, $\{f_1, f_2\}$, and four model features, $\{m_1, \ldots m_4\}$. Each level represents all possible matches between an image features f_i and a model feature m_j. Therefore, the node m_j on level i represents the match (f_i, m_j).

Camera pose estimation
Constraint analysis

$$H = \sum_{i=1}^{N} \mathbf{n}_{ci}\mathbf{n}_{ri}^T = U\Lambda V^T = \lambda_1 u_1 v_1^T + \lambda_2 u_2 v_2^T + \lambda_3 u_3 v_3^T$$

objective function is

$$J = J_R + J_t = \sum_i \|\mathbf{n}_{ri} - R\cdot\mathbf{n}_{ci}\| + \sum_i [d_{ri} - (d_{ci} + \mathbf{n}_{ri}^T t)]^2$$

define $w = [w_\phi, w_t]$, for the 6-DoF transformation.
the variations caused by w_ϕ and w_t are

$$\Delta J_R = \frac{\partial J_R}{\partial w_\phi}\Delta w_\phi = 2\sum(\mathbf{n}_{ri}\times\mathbf{n}_{ci})^T\Delta w_\phi = 2\sum(\Delta w_R \times \mathbf{n}_{ri})^T\mathbf{n}_{ci}$$

$$\Delta J_t = \frac{\partial J_t}{\partial w_t}\Delta w_t = 2\sum(d_{ci} - d_{ri})\cdot\mathbf{n}_{ri}\cdot\Delta w_t$$

Camera pose estimation

Constraint analysis

the scatter matrix for w_ϕ and w_t are

$$C_R = \sum \frac{\partial J_{Ri}}{\partial w_\phi} \frac{\partial J_{Ri}}{\partial w_\phi}^T = 4\sum (\mathbf{n}_{ri} \times \mathbf{n}_{ci})(\mathbf{n}_{ri} \times \mathbf{n}_{ci})^T$$

$$C_t = \sum \frac{\partial J_{ti}}{\partial w_t} \frac{\partial J_{ti}}{\partial w_t}^T = 4\sum (d_{ri} - d_{ci})^2 \cdot \mathbf{n}_{ri}\mathbf{n}_{ri}^T$$

the whole scatter matrix for w is

$$C = \frac{\partial J_i}{\partial w} \frac{\partial J_i}{\partial w}^T = 4\sum \begin{bmatrix} (\mathbf{n}_{ri} \times \mathbf{n}_{ci})(\mathbf{n}_{ri} \times \mathbf{n}_{ci})^T & \mathbf{0_{3\times 3}} \\ \mathbf{0_{3\times 3}} & (d_{ri} - d_{ci})^2 \cdot \mathbf{n}_{ri}\mathbf{n}_{ri}^T \end{bmatrix}$$

then

$$\Delta J^2 = \Delta w C \Delta w$$

Camera pose estimation
Constraint analysis

- 6-DoF case

  the transformation R and t can be determined by plane parameters.

- 5-DoF case

  R and $t_1 + t_2$, where $t_2 = \mu v_3$

  R and t_1 can be determined by plane parameters.

  $\lambda_3 = 0$, $\mathbf{n}_{ci}^T u_3 = 0$, $\mathbf{n}_{ri}^T v_3 = 0$

$$\begin{bmatrix} 0 \\ v_3 \end{bmatrix}^T C \begin{bmatrix} 0 \\ v_3 \end{bmatrix} = 0$$

  so if $w = \alpha[0, v_3^T]^T$ then $\Delta J^2 = 0$

  i.e. if w is along $w = [0, v_3^T]^T$, it will have no impact on J.

- 3-DoF case

  $\lambda_2 = \lambda_3 = 0$

  $(\mathbf{n}_{ri} \times \mathbf{n}_{ci})^T \cdot v_1 = (v_1 \times \mathbf{n}_{ri})^T \cdot \mathbf{n}_{ci} = 0$

  so if $w = [v_1^T, \alpha v_2^T + \beta v_3^T]^T$, then $\Delta J^2 = 0$

Camera pose estimation

Plane filtering

$$F(u,v) = C_{depth}(u,v)[Col(u,v) + C_{center}(u,v) \cdot Edg(u,v)]$$

where

- $C_{depth}(u,v) = k_1 \frac{1}{\sigma_d^2}$ [K. Khoshelham, Sensors, 2012]
- $C_{center}(u,v) = k_2 \left[|u - c_x| + |v - c_y|\right]$
- $Col(u,v) = g(u,v) \otimes c(u,v)$
- $Edg(u,v) = d(u,v) \otimes e(u,v)$, where $c(u,v)$ and $e(u,v)$ are convolution kernel

Camera pose estimation

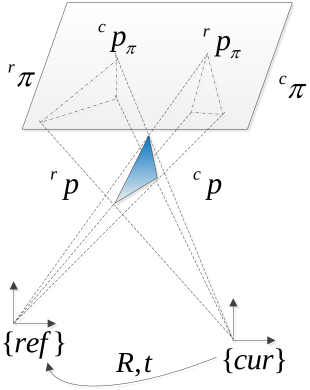
Shadows on plane

- in current frame:
  - plane ${}^c\pi = [{}^c\mathbf{n}^T, {}^c d]^T$
  - projecting ${}^c p$ onto plane ${}^c\pi$ yields

$$ {}^c p_\pi = \frac{{}^c d}{{}^c\mathbf{n}^T\,{}^c p}\,{}^c p \qquad (1)$$

- in reference frame:

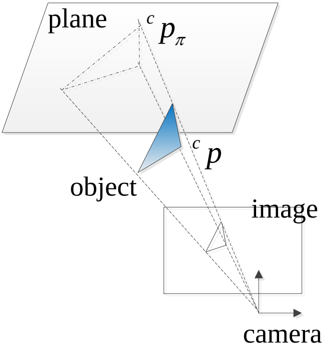
$$ {}^r p_\pi = \frac{{}^r d}{{}^r\mathbf{n}^T\,{}^r p}\,{}^r p \qquad (2)$$



Camera pose estimation

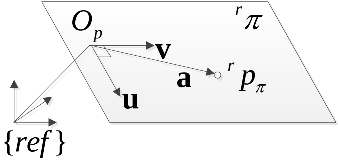
Shadows on plane

- Outline
  - filter the shadow boundaries ${}^{c}p_{\pi}$ and ${}^{r}p_{\pi}$ on planes for two frames.
  - project the boundary points to image, and find the corresponding “object” boundaries ${}^{c}p$ in 3d camera frame that cause the shadows on planes.
  - transform ${}^{c}p$ using R, t yielding ${}^{r}p' = T({}^{c}p)$.
  - project ${}^{r}p'$ to planes via (2) yielding ${}^{r}p'_{\pi}$.
  - align ${}^{r}p'_{\pi}$ with ${}^{r}p_{\pi}$.



Shadows on plane

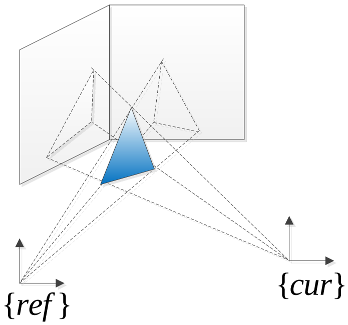
- Alignment of ${}^r p'_\pi$ and ${}^r p_\pi$.
  - define 2d coordinate system on plane ${}^r\pi$.
    - origin: $O_\pi = -{}^r d \cdot {}^r\mathbf{n}$.
    - axis: any two orthogonal unit vectors $\mathbf{u}, \mathbf{v}$ on ${}^r\pi$.
  - transformation from 3d camera coordinate to 2d plane coordinate.
    - 2d coordinate $q_\pi = [u, v]^T$
    - $u = \mathbf{a}^T \mathbf{u}$
    - $v = \mathbf{a}^T \mathbf{v}$
    - where $\mathbf{a}$ is the vector from O_p to ${}^r p_\pi$
  - PMICP [R. Guo, 2009]

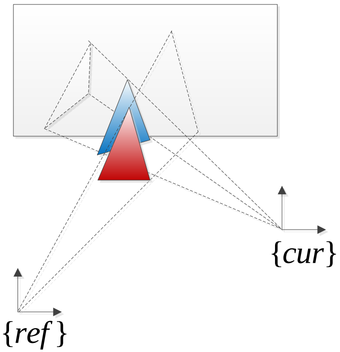


Camera pose estimation

Shadows on plane

- Some drawbacks



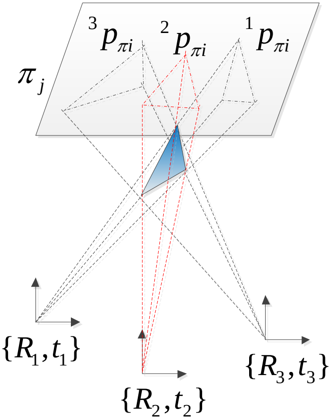


Loop closing and map optimization

Loop closing

Loop closing and map

Map optimization



Reference

K. Khoshelham and S. O. Elberink, “Accuracy and resolution of Kinect depth data for indoor mapping applications,” Sensors, vol. 12, no. 2, pp. 1437-1454, 2012.

R. Guo, F. Sun and J. Yuan, “ICP based on polar point matching with application to Graph-SLAM,” in Proc. Mechatronics and Automation, 2009.