Notes on Visual SLAM 14 lectures – Chapter 12

1 Mapping in SLAM

A. localization: visual odometry or re-location with the descriptors, only need to map once instead of doing a complete SLAM every time the robot is started

B. Navigation: the process in which a robot can plan a path on a map, find a path between any two map points, and then control its movement to a target point, need to know passable places on the map

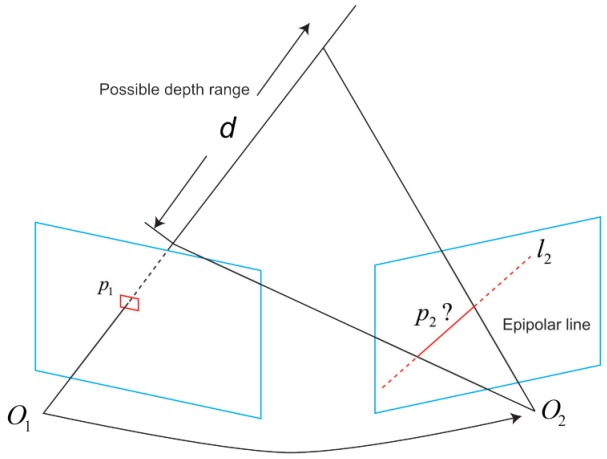
C. obstacle avoidance: similar to navigation but pays more attention to the handling of local and dynamic obstacles.

D. reconstruction: mostly for demonstration

E. interaction: human-robot interaction, high-level commands to robots regarding actions in the map, requires robots to have a higher level of knowledge of semantic maps

2 Monocular dense reconstruction

depth measured by monocular and binocular is often fragile. Using RGB-D is more reliable choice. Which



point on the epipolar line is the p_1 point we just saw?
Take a small block of size $w \times w$ around p_1 and then take many small blocks of the same size on the epipolar line for comparison, which is the block-match method. Denote the small block around p_1 as $A \in \mathbb{R}^{w \times w}$, and denote the n small blocks on the epipolar line into $B_i, i = 1, ..., n$. There are several methods for calculating the difference between a small block and another block;

A. SAD(sum of absolute differences):

$$S(A, B)_{SAD} = \sum_{i,j} |A(i, j) - B(i, j)| \tag{1}$$

B. SSD(sum of squared distance):

$$S(A,B)_{SSD} = \sum_{i,j}(A(i,j) - B(i,j))^2 \tag{2}$$

C. NCC(normalized cross correlation):

$$S(A,B)_{NCC} = \frac{\sum_{i,j} A(i,j)B(i,j)}{\sqrt{\sum_{i,j} A(i,j)^2 \sum_{i,j} B(i,j)^2}} \tag{3}$$

correlation close to 0 means two images are not similar, and close to 1 means similar.

Use probability distributions to describe depth values rather than using a single value to describe the depth.

2.1 Gaussian Depth Filters

Assume the depth d of a certain pixel satisfy:

$$P(d) = N(\mu, \sigma^2) \tag{4}$$

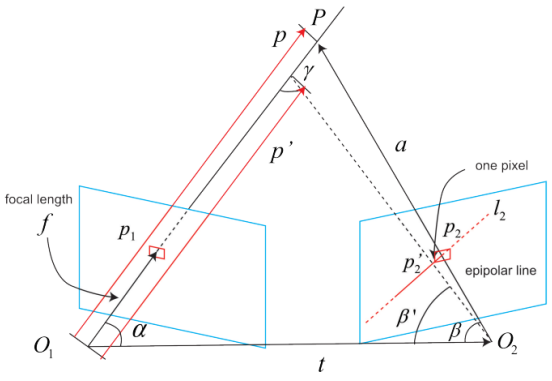
whenever new data arrives, we will observe its depth. Similarly, suppose this observation is also a Gaussian distribution:

$$P(d_{obs}) = N(\mu_{obs}, \sigma_{obs}^2) \tag{5}$$

Suppose the distribution of d after fusion is $N(\mu_{fuse}, \sigma_{fuse}^2)$, then according to the product of the Gaussian distribution, we have:

$$\mu_{fuse} = \frac{\sigma_{obs}^2 \mu + \sigma^2 \mu_{obs}}{\sigma^2 + \sigma_{obs}^2}, \sigma_{fuse}^2 = \frac{\sigma^2 \sigma_{obs}^2}{\sigma^2 + \sigma_{obs}^2} \tag{6}$$

list the geometric relationship between these quantities, we have:



$$a = p - t$$
$$\alpha = \arccos\langle p, t\rangle$$
$$\beta = \arccos\langle a, -t\rangle$$

Perturbing p_2 by one pixel will cause β to produce a change, which becomes β', according to the geometric relationship, there are:

$$\beta' = \arccos\langle O_2 p_2', -t\rangle \tag{7}$$

and

$$\gamma = \pi - \alpha - \beta' \tag{8}$$

Therefore, according to the law of sine $\frac{\sin(A)}{A} = \frac{\sin(B)}{B} = \frac{\sin(C)}{C}$, p' can be written as:

$$||p'|| = ||t|| \frac{\sin \beta'}{\sin \gamma} \tag{9}$$

thus, we compute the depth uncertainty caused by the uncertainty of a single pixel:

$$\sigma_{obs} = ||p|| - ||p'|| \tag{10}$$

steps for estimating the pixel depth:

A. Assume that the depth of all pixels meets an initial Gaussian distribution

B. when a new image is generated, the projected point's location is determined through epipolar search and block matching

C. Calculate the depth and uncertainty of the triangle based on the geometric relationship

D. Fuse the current observation into the last estimate, if it converges, stop the calculation, otherwise return to step 2

2.2 Discussion

A. for block matching, we must assume that the small block is unchanged, and then we compared it with other blocks

B. blocks with conspicuous gradients (obvious texture) can achieve better results

C. when epipolar line is orthogonal to the pixel gradient, no effective matching can be obtained; when epipolar line is parallel to the pixel gradient, matching can be accurately obtained

2.2.1 Inverse depth filter

Assume the depth value satisfies the Gaussian distribution: $d \sim N(\mu, \sigma^2)$, there are some issues with Gaussian distribution for depth:

A. the distribution does not form a symmetrical shape like the Gaussian distribution, its tail may be slightly longer, and the negative area is zero.

B. there may be points very far away in some outdoor applications or even at infinity, which is difficult to cover

Inverse depth Gaussian: more effective, better numerical stability

2.2.2 Pre-transform the image

A point P_R in the reference frame has the following relationship with the 3D point P_W:

$$d_R P_R = K(R_{RW} P_W + t_{RW}) \tag{11}$$

and similarly, for the current frame, there is a projection of P_W on it:

$$d_C P_C = K(R_{CW} P_W + t_{CW}) \tag{12}$$

substitute and eliminate P_W, the pixel relationship between the two images is obtained:

$$\begin{aligned}
K R_{RW} P_W &= d_R P_R - K t_{RW} \\
P_W &= (K R_{RW})^{-1} (d_R P_R - K t_{RW}) \\
&= R_{RW}^{-1} K^{-1} d_R P_R - R_{RW}^{-1} K^{-1} K t_{RW} \\
&= R_{RW}^{T} K^{-1} d_R P_R - R_{RW}^{T} t_{RW} \\
d_C P_C &= K R_{CW} (R_{RW}^{T} K^{-1} d_R P_R - R_{RW}^{T} t_{RW}) + K t_{CW} \\
&= d_R K R_{CW} R_{RW}^{T} K^{-1} P_R + K t_{CW} - K R_{CW} R_{RW}^{T} t_{RW}
\end{aligned}$$

when we know d_R, P_R, we can calculate the projection position of P_C. Give two components of P_R an increment d_u, d_v, then the increment of P_C can be obtained as du_c, dv_c:

$$\begin{bmatrix} du_c \\ dv_c \end{bmatrix} = \begin{bmatrix} \frac{du_c}{du} & \frac{du_c}{dv} \\ \frac{dv_c}{du} & \frac{dv_c}{dv} \end{bmatrix} \begin{bmatrix} du \\ dv \end{bmatrix} \tag{13}$$

we can transform the current frame's pixels and then perform block matching to obtain a better effect on rotation.

2.2.3 Other improvements

- to ensure smoothness on the depth map, we can assume that the adjacent depth will not change much, and add a spatial regularization term to the depth estimation

2.3 RGB-D mapping

A. Cannot be directly used for localization

B. Cannot be directly used for navigation and obstacle avoidance, needs postprocessing to obtain a map format that is more suitable

C. Does not conform to people's visualization habits

2.3.1 Surfel mapping

First calculate the point cloud's normal and then calculate the grid from the normal

2.3.2 Octo-mapping

Octa tree saves space since when all the child nodes of a block are not occupied, we do not need to expand the node. Since occupied objects and blank spaces are often connected together, most octree nodes do not need to be expanded to the leaf node.
Use 0 for blank and 1 for occupied, initialize all nodes to 0.5 at the beginning. Let $y \in \mathbb{R}$ be the logarithmic value and x be the probability of $0 \sim 1$, then we have the logit transformation:

$$y = \text{logit}(x) = \log(\frac{x}{1-x}) \tag{14}$$

and the inverse transform is:

$$x = \text{logit}^{-1}(y) = \frac{\exp(y)}{\exp(y)+1} \tag{15}$$

in which when y takes 0, x takes 0.5. Suppose a certain node is n, and the observed data is z, then the logarithmic value of the probability of a node from the beginning to the moment t is $L(n|z_{1:t})$, and the time $t+1$ is:

$$L(n|z_{1:t+1}) = L(n|z_{1:t-1}) + L(n|z_{1:t}) \tag{16}$$

in which $L(n) = \log(\frac{P(n)}{1-P(n)})$, the probabilistic form is:

$$P(n|z_{1:T}) = \left[1 + \frac{1-P(n|z_T)}{P(n|z_T)} \frac{1-P(n|z_{1:T-1}}{P(n|z_{1:T-1}} \frac{P(n)}{1-P(n)}\right]^{-1} \tag{17}$$

Suppose we observe a specific pixel with depth d in the RGB-D image, it means

- The observed point is occupied
- The line from the camera center to the observed point is free

2.3.3 TSDF

Truncated-signed distance function: if the block is in front of the object's surface, it has a positive value; if it is behind the surface, it has a negative value. The place where TSDF changes from negative to positive is the surface itself.

3 Appendix

The general problem of mapping:

- Given the sensor data
$$d = \{u_1, z_1, u_2, z_2, ..., u_t, z_t\} \tag{18}$$
- Calculate the most likely map:
$$m^* = \arg\max_m P(m|d) \tag{19}$$

Assumption 1: The area that corresponds to a cell is either completely free or occupied, each cell is a binary random variable that models the occupancy:
Cell is occupied: $p(m_i) = 1$
Cell is not occupied: $p(m_i) = 0$
No information: $p(m_i) = 0.5$
The probability distribution of the map is given by the product of the probability distributions of the individual cell:
$$p(m) = \prod_i p(m_i) \tag{20}$$

Given a sensor data $z_{1:t}$ and the poses $x_{1:t}$ of the sensor, estimate the map:
$$p(m|z_{1:t}, x_{1:t}) = \prod_i p(m_i|z_{1:t}, x_{1:t}) \tag{21}$$

We can write $p(m|z_{1:t}, x_{1:t})$ as:

$$\begin{aligned}
p(m_i|z_{1:t}, x_{1:t}) &= p(m_i|z_t, z_{1:t-1}, x_{1:t}) \\
&= \frac{P(m_i, z_t|z_{1:t-1}, x_{1:t})}{P(z_t|z_{1:t-1}, x_{1:t})} \\
&= \frac{\frac{P(m_i, z_t, z_{1:t-1}, x_{1:t})}{P(z_{1:t-1}, x_{1:t})}}{P(z_t|z_{1:t-1}, x_{1:t})} \\
&= \frac{\frac{P(m_i, z_t, z_{1:t-1}, x_{1:t})}{P(z_{1:t-1}, x_{1:t})}}{P(z_t|z_{1:t-1}, x_{1:t})} \\
&= \frac{\frac{P(z_t|m_i, z_{1:t-1}, x_{1:t})P(m_i, z_{1:t-1}, x_{1:t})}{P(z_{1:t-1}, x_{1:t})}}{P(z_t|z_{1:t-1}, x_{1:t})} \\
&= \frac{\frac{P(z_t|m_i, z_{1:t-1}, x_{1:t})P(m_i|z_{1:t-1}, x_{1:t})P(z_{1:t-1}, x_{1:t})}{P(z_{1:t-1}, x_{1:t})}}{P(z_t|z_{1:t-1}, x_{1:t})} \\
&= \frac{P(z_t|m_i, z_{1:t-1}, x_{1:t})P(m_i|z_{1:t-1}, x_{1:t})}{P(z_t|z_{1:t-1}, x_{1:t})}
\end{aligned}$$

According to the Markov property, the above expression can be written as:
$$p(m_i|z_{1:t}, x_{1:t}) = \frac{P(z_t|m_i, x_t)P(m_i|z_{1:t-1}, x_{1:t})}{P(z_t|z_{1:t-1}, x_{1:t})} \tag{22}$$

According to Bayes rule, the first term in the numerator can be written as:

$$P(z_t|m_i, x_t) = \frac{P(m_i|z_t, x_t)P(z_t|x_t)}{P(m_i|x_t)} \tag{23}$$

plug that into equation 22, we have:

$$p(m_i|z_{1:t}, x_{1:t}) = \frac{P(m_i|z_t, x_t)P(z_t|x_t)P(m_i|z_{1:t-1}, x_{1:t})}{P(m_i|x_t)P(z_t|z_{1:t-1}, x_{1:t})}$$

According to Markov property, the map is only updated by the observation, instead of the pose of the sensor:

$$p(m_i|z_{1:t}, x_{1:t}) = \frac{P(m_i|z_t, x_t)P(z_t|x_t)P(m_i|z_{1:t-1}, x_{1:t})}{P(m_i)P(z_t|z_{1:t-1}, x_{1:t})} \tag{24}$$

Do exactly the same for the opposite event:

$$p(\neg m_i|z_{1:t}, x_{1:t}) = \frac{P(\neg m_i|z_t, x_t)P(z_t|x_t)P(\neg m_i|z_{1:t-1}, x_{1:t})}{P(\neg m_i)P(z_t|z_{1:t-1}, x_{1:t})} \tag{25}$$

Then the ratio of both probabilities becomes:

$$\frac{p(m_i|z_{1:t}, x_{1:t})}{p(\neg m_i|z_{1:t}, x_{1:t})} = \frac{\frac{p(m_i|z_t, x_t)P(m_i|z_{1:t-1}, x_{1:t})}{P(m_i)}}{\frac{p(\neg m_i|z_t, x_t)P(\neg m_i|z_{1:t-1}, x_{1:t})}{P(\neg m_i)}} \tag{26}$$

Rearranging, we have:

$$\begin{aligned}\frac{P(m_i|z_{1:t}, x_{1:t})}{1 - P(m_i|z_{1:t}, x_{1:t})} &= \frac{P(m_i|z_t, x_t)}{P(\neg m_i|z_t, x_t)} \frac{P(m_i|z_{1:t-1}, x_{1:t})}{P(\neg m_i|z_{1:t-1}, x_{1:t})} \frac{P(\neg m_i)}{P(m_i)} \\ &= \underbrace{\frac{p(m_i|z_t, x_t)}{1 - P(m_i|z_t, x_t)}}_{\text{uses } z_t} \underbrace{\frac{p(m_i|z_{1:t-1}, x_{1:t})}{1 - P(m_i|z_{1:t-1}, x_{1:t})}}_{\text{recursive term}} \underbrace{\frac{1 - P(m_i)}{P(m_i)}}_{\text{prior}}\end{aligned}$$

Let us denote $P(m_i|z_{1:t}, x_{1:t})$ as x, and right side of equation as y, then we have:

$$\frac{x}{1 - x} = y \tag{27}$$

rearranging:

$$\begin{aligned} x &= y - xy \\ x(1 + y) &= y \\ x &= \frac{y}{1 + y} \\ &= \frac{1}{1 + \frac{1}{y}} \end{aligned}$$

Then we have:

$$P(m_i|z_{1:t}, x_{1:t}) = \left[1 + \frac{1 - P(m_i|z_t, x_t)}{P(m_i|z_t, x_t)} \frac{1 - P(m_i|z_{1:t-1}, x_{1:t})}{P(m_i|z_{1:t-1}, x_{1:t})} \frac{P(m_i)}{1 - P(m_i)}\right]^{-1} \tag{28}$$

which corresponds with equation 17.
Define log odds ratio as:

$$l(x) = \log \frac{P(x)}{1 - P(x)} \tag{29}$$

We then have:

$$l(m_i|z_{1:t}, x_{1:t}) = \underbrace{l(m_i|z_t, x_t)}_{\text{inverse sensor model}} + l(m_i|z_{1:t-1}, x_{1:t-1}) - l(m_i) \tag{30}$$

or in short form:

$$l_{t,i} = \text{inv-sensor-model}(m_i, x_t, z_t) + l_{t-1,i} + l_0 \tag{31}$$