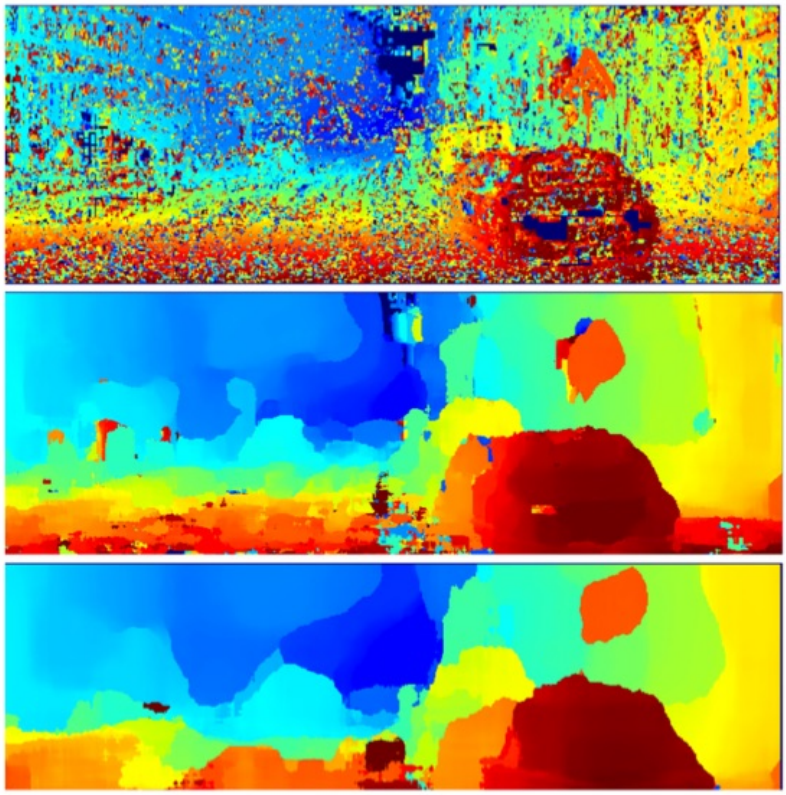
LECTURE 10 : DEPTH FROM STEREO & VIDEO

[1] Disparity Map from Rectified Images

Pixel or patch based operations :



patch size = 5

patch size = 35

patch size = 85

Problem : small patches → more details but more noise
large patch → less noise but less details

Solution : MRF

$$D^* = \underset{\{D\}}{\operatorname{argmin}} \sum_x \left(f_d(x, D(x), I, I') + \lambda \sum_{y \in N_x} f_p(D(x), D(y)) \right)$$

- D^* : disparity map
- $\{D\}$: $\{D\} = \{ d_{min} \dots d_{max} \}$ the range of disparity values
- x : pixel location
- $f_d(x, D(x), I, I')$: disparity to estimate
- I, I' : input images

Data term:

$$f_d(x, D(x), I, I') = \left[I(x) - I'(x + D(x)) \right]^2$$

Photo-consistency constraint

If $\{d_{min} \ldots d_{max}\}$ has 10 values: $d_1, d_2, \ldots d_{10}$; then pixel x has 10 possible values of fd, where each of them is the data cost of d_i.

Prior term:

$$f_p(D(x), D(y)) = (D(x) - D(y))^2$$

smoothness constraint

If we have 10 values of $\{d_{min} .. d_{max}\}$ for each pixel, we will have 100 values of fp:

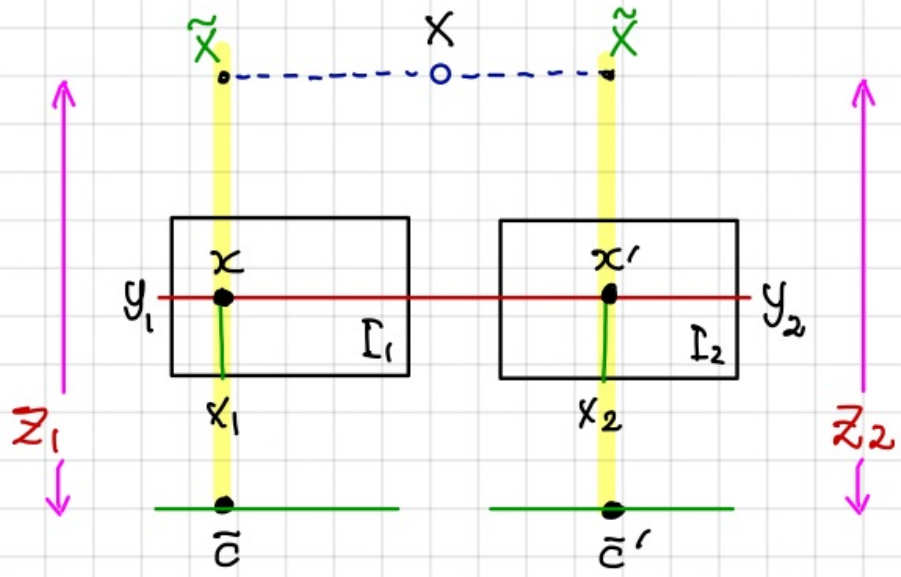
D(y) \ D(x)	d_1	d_2	...	d_{10}
d_1	$(d_1-d_1)^2$	$(d_2-d_1)^2$		$(d_{10}-d_1)^2$
d_2	⋮	⋮		⋮
⋮				
d_{10}				

These values are applied to all pixels, exactly in the same way.

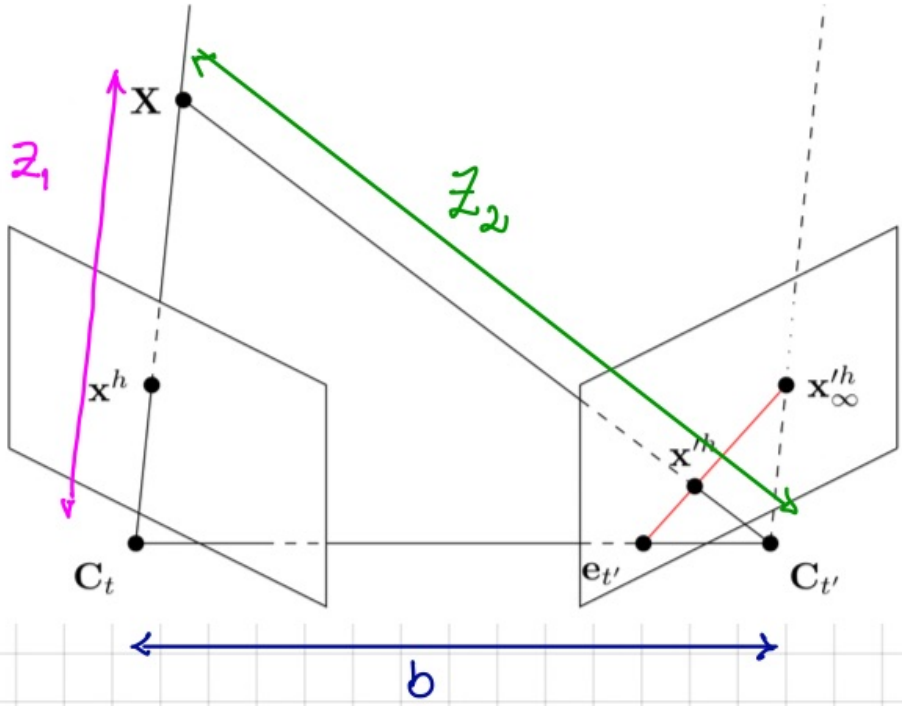
↓↓

Having set the data and prior term for every pixel of the left image, we optimize the graph using graphcuts.

For a pair of rectified images, there is only one disparity or depth map:



Though, before being rectified, there are 2 depth maps:



Q: How to compute disparity of non-rectified images?

A: Two ways:

(1) Rectify the images → Epipolar rectification

(2) Using epipolar geometry

⇓

Epipolar-geometry implicit rectification is preferable, since less steps are required in the computation.

↓

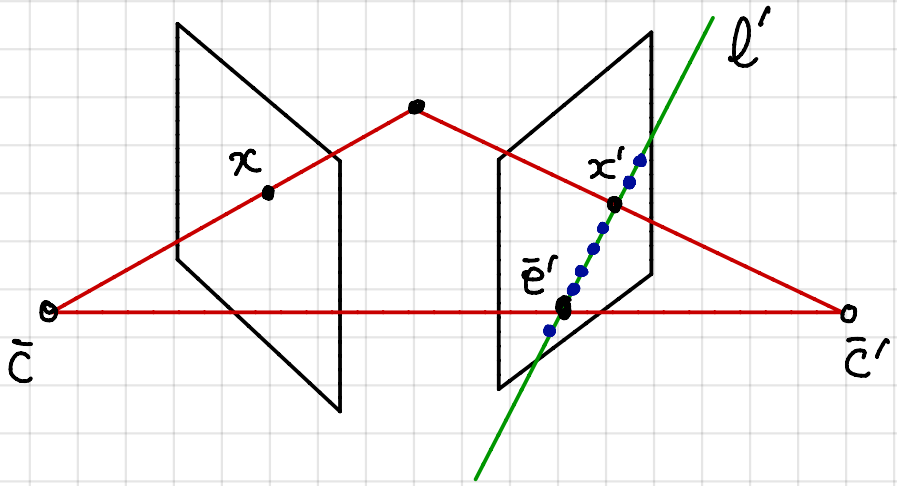
The basic formulation:

$d = \frac{1}{z}$; d = disparity, z = depth

$$x' \sim K' R' R^T K^{-1} x + d\, K' R' (\bar{C} - \bar{C}')$$

Given x, $P = K[R|t]$ and $P' = K'[R'|t']$, we want to find d using the formula, such that $I(x) = I'(x')$.

The formula can generate all points on the green line below, when we change the value of d:

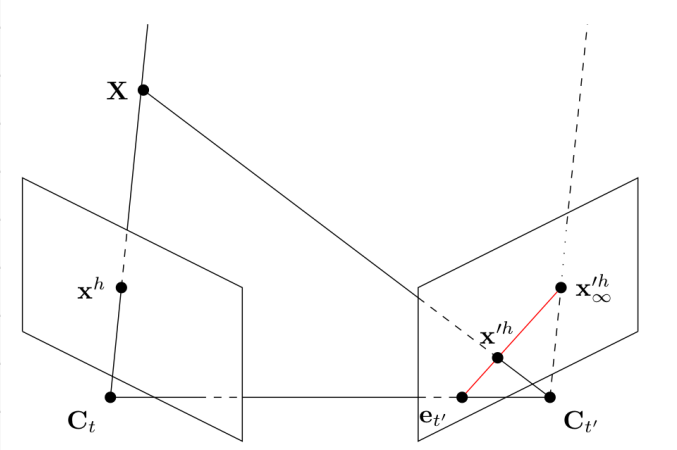


[4] Derivation of Epipolar Rectification

To understand how we can get this equation:

$$x' \sim K' R' R^T K^{-1} x + d K' R' (\bar{C} - \bar{C}').$$

consider:



$$x^h = PX = K[R|\bar{t}]X$$

Assume the existence of X_∞ along the line that passes through X, where X_∞ is located at infinity:

$$X_\infty = \begin{bmatrix} X \\ Y \\ Z \\ 0 \end{bmatrix} = \begin{bmatrix} \hat{X}_\infty \\ 0 \end{bmatrix}$$

$$\underset{3\times1}{x^h} = \underset{3\times3}{K}\underset{3\times3}{[R}\ \underset{3\times1}{\bar{t}]}\ \underset{4\times1}{X_\infty} = \underset{3\times3}{K}\underbrace{[R\ \bar{t}]}_{3\times4}\underset{4\times1}{\begin{bmatrix} \hat{X}_\infty \\ 0 \end{bmatrix}} = \underset{3\times3}{K}\ \underset{3\times3}{R}\ \underset{3\times1}{\hat{X}_\infty}$$

Thus:

$$\hat{X}_\infty = R^T K^{-1} x^h$$

Projecting X_∞ onto the right image:

$$x'^h_\infty = P' X_\infty = K'[R'|\bar{t}'] \begin{bmatrix} \hat{X}_\infty \\ 0 \end{bmatrix} = K' R' \hat{X}_\infty$$

$$= K' R' R^T K^{-1} x^h$$

What we want to find is x'^h, and not x'^h_∞.

Basic idea of finding x'^h:

To find x'^h, we can start the search from x'^h_∞. Then from x'^h_∞, we can trace along the epipolar line in the direction of the epipolar point, $e_{t'}$:

$$\underset{3\times1}{\bar{e}_{t'}} = \underset{3\times4}{P'}\ \underset{4\times1}{\bar{C}_t} = K'[R'|\bar{t}']\,\bar{C}_t$$

$$= K'R'[\underset{3\times3}{I}\ |\ \underset{3\times1}{[R']^{-1}\bar{t}'}]\ \underset{4\times1}{\bar{C}_t}$$

$$= \underset{3\times3}{K'}\ \underset{3\times3}{R'}[\underset{3\times1}{\bar{C}_t} - \underset{3\times1}{\bar{C}_{t'}}]$$

Recall: $\bar{C}_{t'} = -(R')^{-1}\bar{t}'$

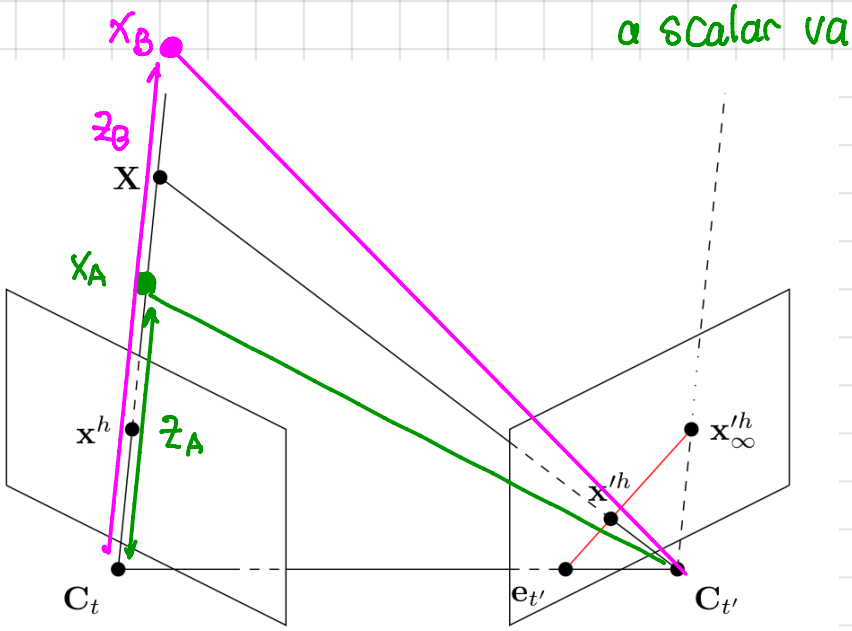
Therefore:

$$\underset{3\times1}{x'^h} \sim \underset{3\times1}{x'^h_\infty} + d\ \underset{3\times1}{\bar{e}_{t'}}$$ → the line parametric equation

$$x'^h \sim K'R'R^TK^{-1}x^h + d\,\bar{e}_{t'}$$

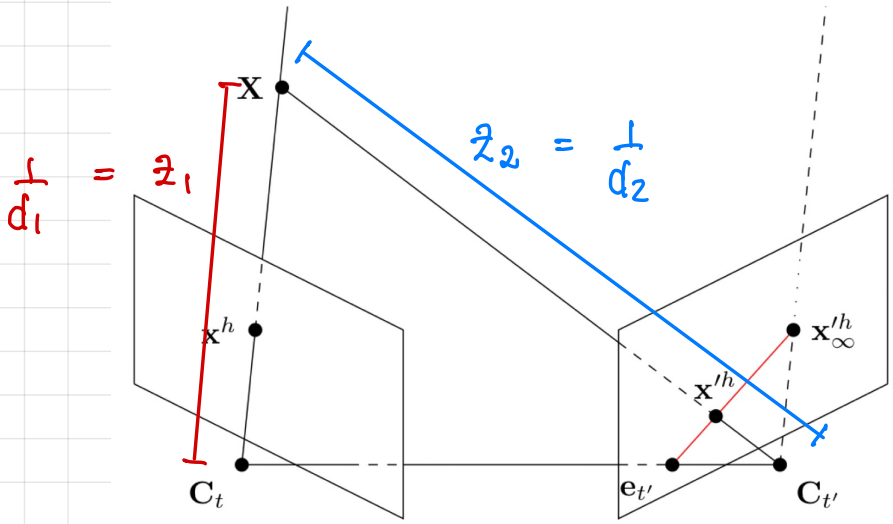
$$\underset{3\times1}{x'^h} \sim \underset{3\times3}{K'}\ \underset{3\times3}{R'}\ \underset{3\times3}{R^T}\ \underset{3\times3}{K^{-1}}\ \underset{3\times1}{x^h} + d\ \underset{3\times3}{K'}\ \underset{3\times3}{R'}[\underset{3\times1}{\bar{C}_t} - \underset{3\times1}{\bar{C}_{t'}}]$$

d: a scalar value



The shorter the depth (Z_A), the larger the value of d, vice versa.

This also indicates d determines the depth of point X.



d = disparity
z = depth

In the figure, with respect to X, there are 2 depths:

- $z_1 = 1/d_1$ is the depth from C_t to X
- $z_2 = 1/d_2$ is the depth from $C_{t'}$ to X

$$x'^h \sim x'^h_\infty + d_1 e_{t'}$$

3x1 3x1 3x1

→ You need to normalize x'^h so that: $\begin{pmatrix} x \\ y \\ z \end{pmatrix} \rightarrow \begin{pmatrix} x/z \\ y/z \\ 1 \end{pmatrix}$

Q: How to justify that this equation is correct?

A: Consider the following cases:

If $d = 0$, meaning $z = \infty$:

$$x'^h = x'^h_\infty$$

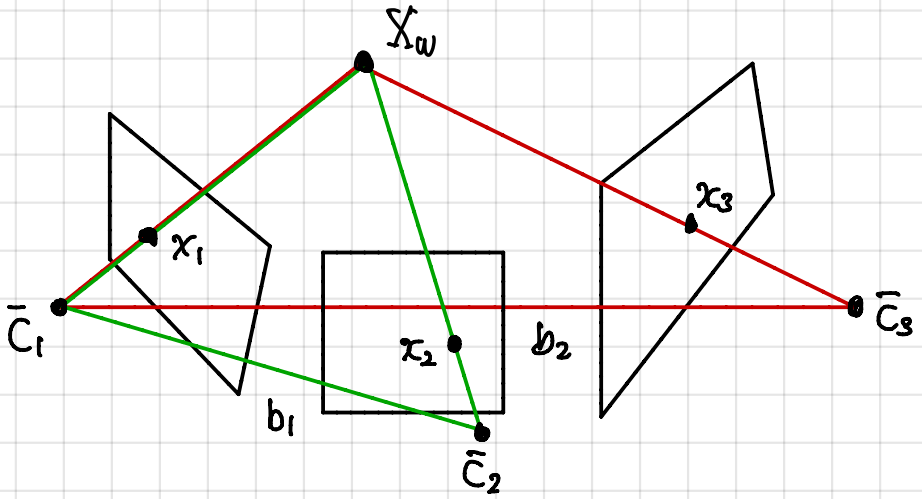
If $d = \infty$, meaning $z = 0$ (the 3D point is at $\bar{C}_t$):

$$x'^h = x'^h_\infty + \infty e' = e'$$

} see the illustration above

If $0 < d < \infty$, then x'^h must lie on the epipolar line!

[5] Epipolar Rectification for Multiple Images



[o] For image 1 & 2 :

$$x_2 = x_{2\infty} + d_{12}\,\bar{e}_2 = K_2 R_2 R_1^T K_1^{-1} x_1 + d_{12} K_2 R_2 (\bar{C}_1 - \bar{C}_2)$$

[o] For image 1 & 3 :

$$x_3 = x_{3\infty} + d_{13}\,\bar{e}_3 = K_3 R_3 R_1^T K_1^{-1} x_1 + d_{13} K_3 R_3 (\bar{C}_1 - \bar{C}_3)$$

$$\Downarrow$$

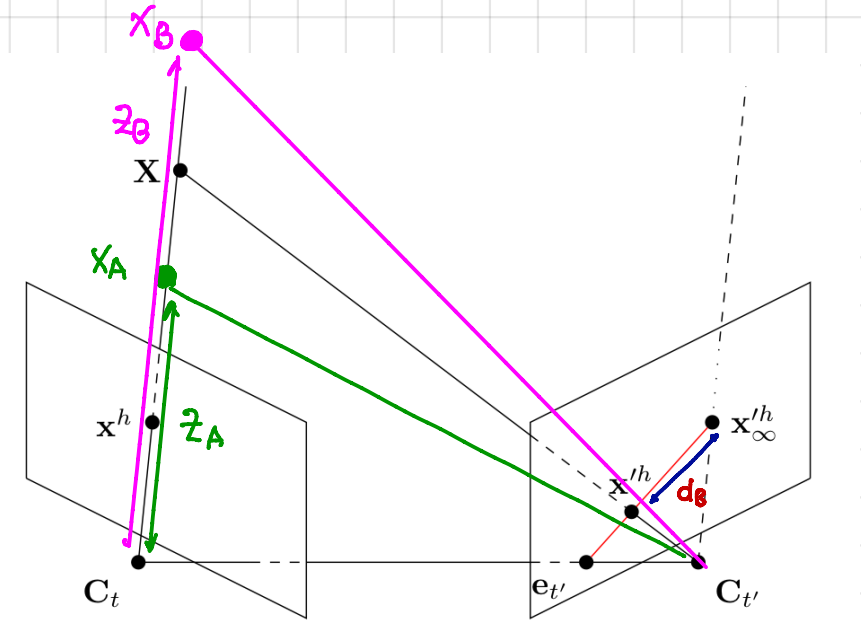
$$\boxed{d_{12} = d_{13}}$$

Implying : $I_2(x_2) = I_1(x_1)$

$I_3(x_3) = I_1(x_1)$

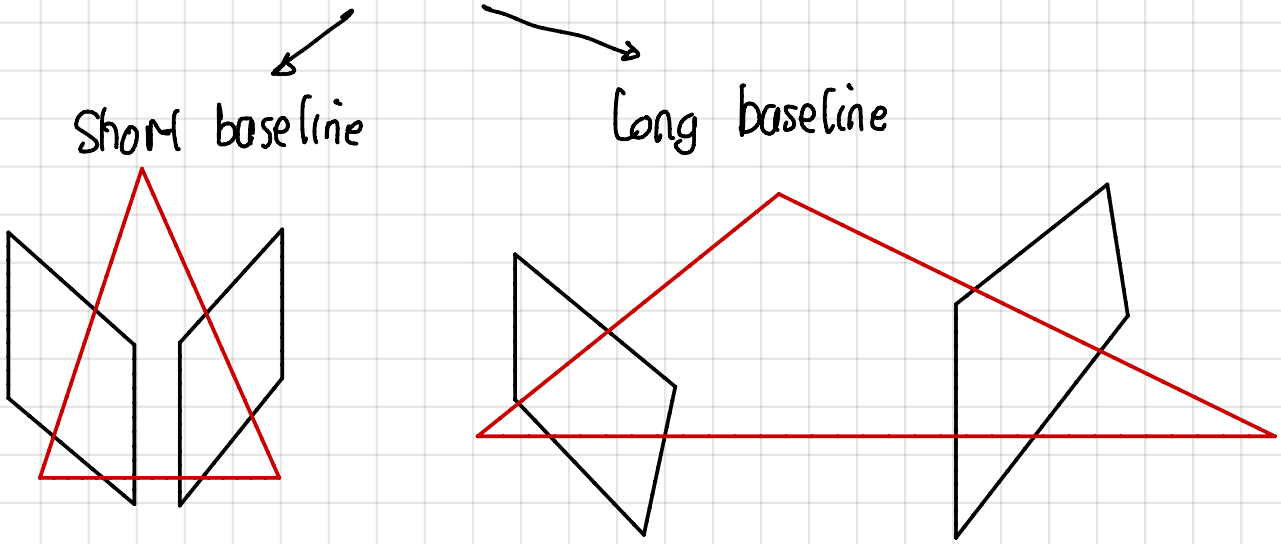
Q : Why $d_{12} = d_{13}$?

A : Because d_{12} & d_{13} indicate the same depth of X from C_1.



[6] Multiple Baseline Stereo = Depth from Video #11

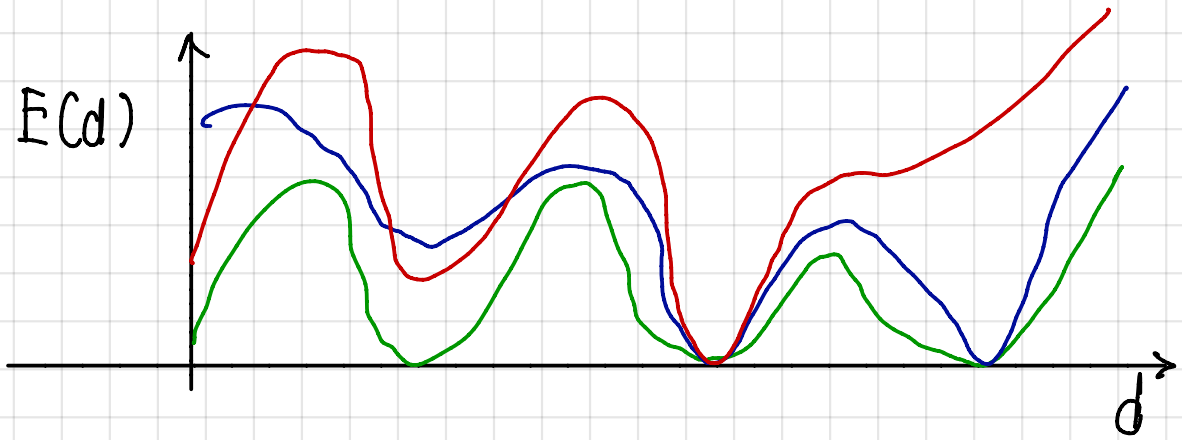
Disparity values depends on the baseline



For estimating disparity, a longer baseline is generally better, since b in the equation acts like a magnifier. However, longer baselines tend to suffer from occlusions.

↓

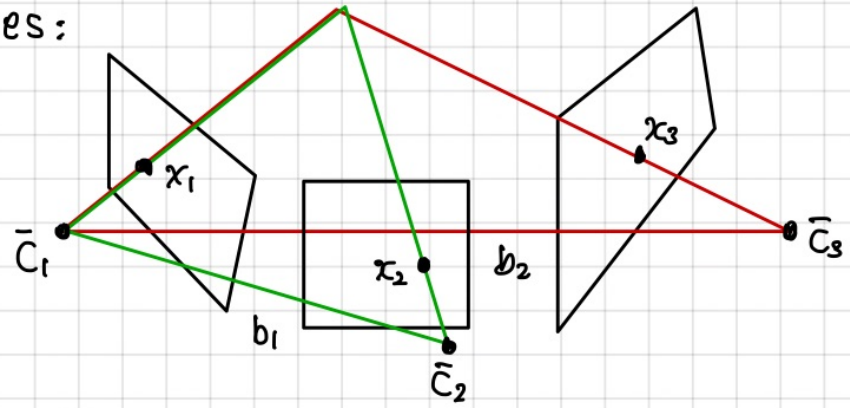
Multiple baselines allow us to get the benefits of short and long baselines:



The green line indicates that there are many candidates of d that make $E(d)=0$ ⇒ ambiguity!

[7] Depth from Video (Multiple Baselines)

Multiple baselines:



Two main steps

(1) Disparity Initialization

(2) Bundle Optimization

Purpose: To have an initial depth map of each frame.

Input: $\{I_t, P_t\}_{t=1}^{N}$; $t = 1 \dots N$

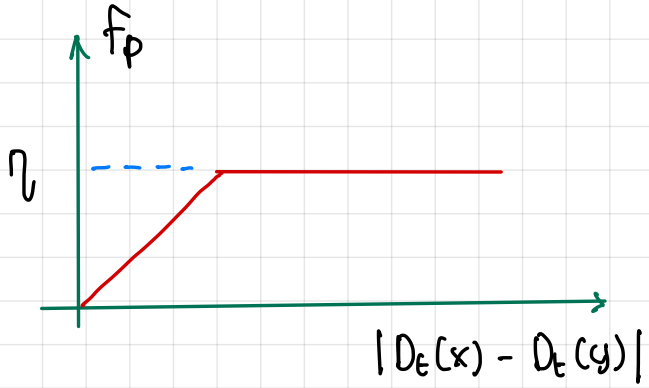
Output: $\{D_t\}_{t=1}^{N}$ (the disparity maps)

N = the number of frames.

MRF:

$$D_t^{init} = \underset{\{d_{min} \dots d_{max}\}}{\operatorname{argmin}} \sum_x \left[(1 - U(x)) f_d^{init}(x, D_t(x)) - \sum_{y \in N_x} \lambda(x,y) f_p(D_t(x), D_t(y)) \right]$$

- Prior term: $f_p(D_t(x), D_t(y)) = \min\left(|D_t(x) - D_t(y)|, \eta\right)$



To prevent f_p from being too large, since the difference between two disparity values shouldn't be that large.

Main reference: "Consistent depth map recovery from a video sequence." TPAMI, 2009.

MRF:

$$D_t^{init} = \underset{\{d_{min} \dots d_{max}\}}{\operatorname{argmin}} \sum_x \left[1 - U(x) f_d^{init}(x, D_t(x)) - \sum_{y \in N_x} \lambda(x,y) f_p(D_t(x), D_t(y)) \right]$$

- Data term:

Use all other frames (prev. & subsequent frames)

$$f_d^{init}(x, D_t(x)) = \sum_{t'}^{N} \rho_c(x, D_t(x), I_t, I_{t'})$$

e.g.: $t = 1$, $t' = 2, 3, \dots, N$

where:

$$\rho_c(x, d, I_t, I_{t'}) = \frac{\sigma_c}{\sigma_c + |I_t(x) - I_{t'}(l_{t,t'}(x,d))|}$$

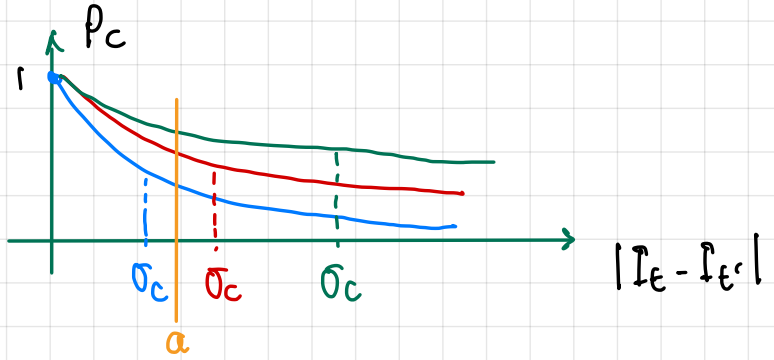
photo consistency

$l_{t,t'}(x_t, d)$ means:

$$x_{t'} = K_{t'} R_{t'} R_t^T K_t^{-1} x_t + d K_{t'} R_{t'} [C_t - C_{t'}]$$

3×1 $\quad 3\times3 \quad 3\times3 \quad 3\times3 \quad 3\times3 \quad 3\times1 \quad 1\times1 \quad 3\times3 \quad 3\times3 \quad 3\times1$

σ_c implies a variable to control the tolerance of the photo consistency:



if $|I_t - I_{t'}| = a$, then different values of σ_c will produce different ρ_c. The smaller σ_c (blue line) penalize $|I_t - I_{t'}|$ more. Meaning, even though $|I_t - I_{t'}|$ is relatively small, ρ_c will be smaller compared with the other values of σ_c (red and green lines).

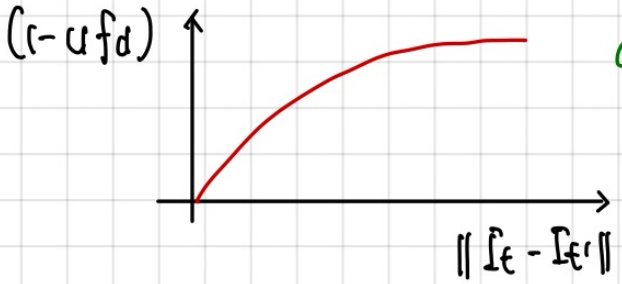
MRF :

$$D_t^{init} = \underset{\{d_{min} \dots d_{max}\}}{\operatorname{argmin}} \sum_x \left[(1 - U(x)) f_d^{init}(x, D_t(x)) - \sum_{y \in N_x} \lambda(x,y) f_p(D_t(x), D_t(y)) \right]$$

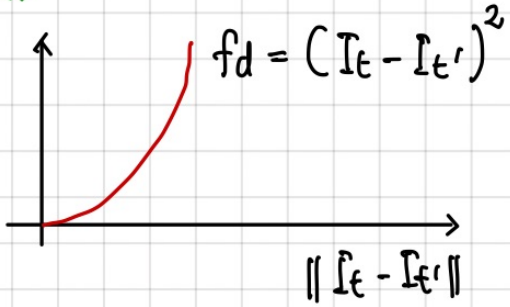
$$U(x) = \frac{1}{\max_{D_t(x)} f_d(x, D(t))}$$

→ the normalization factor. To ensure the values ranging from 0 to 1.

The graph of the overall data term:



more robust than our prev. cost function →



If there is noise $\| I_t - I_{t'} \|$ will be large, hence $(I_t - I_{t'})^2$ will be also large, though the actual cost shouldn't be that large. Using $(1-u f_d)$ will reduce the effect of noise in the optimization.

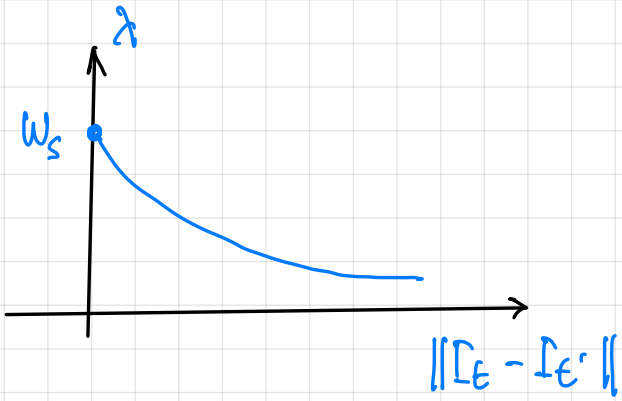
- Weighting factor λ :

MRF :

$$D_t^{init} = \underset{\{d_{min} \dots d_{max}\}}{\operatorname{argmin}} \sum_x \left[1 - U(x) f_d^{init}(x, D_t(x)) - \sum_{y \in N_x} \lambda(x,y) f_p(D_t(x), D_t(y)) \right]$$

Goal : To preserve the discontinuities. λ is defined to encourage the disparity discontinuities to be consistent with intensity / color discontunities.

$$\lambda(x,y) = w_s \frac{U_\lambda(x)}{\| I_t(x) - I_t(y) \| + \varepsilon}$$



where :

$$U_\lambda(x) = \frac{|N_x|}{\sum_{y' N_x} \frac{1}{\| I_t(x) - I_t(y') \| + \varepsilon}}$$

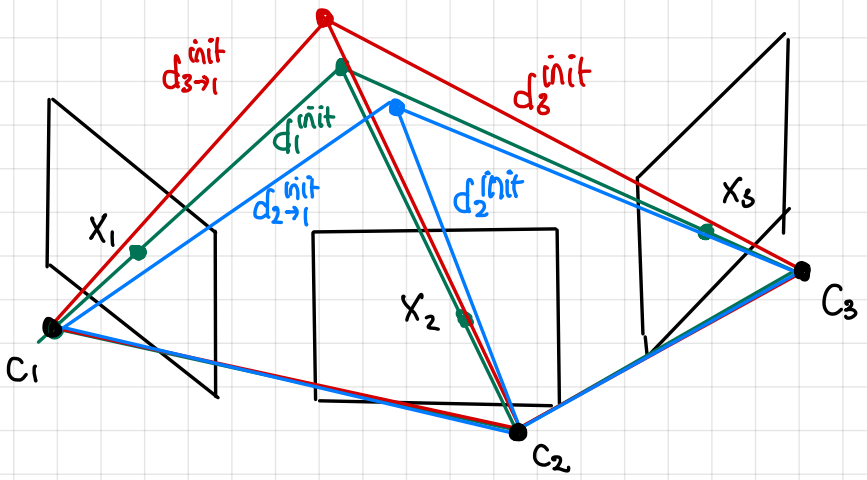
w_s = the smoothness strength

$$\lambda(x,y) = w_s \frac{|N_x|}{(\| I_t(x) - I_t(y) \| + \varepsilon) \sum_{y' \in N_x} \frac{1}{\| I_t(x) - I_t(y') \| + \varepsilon}}$$

1. The smaller the intensity / color difference ($\| I_t(x) - I_t(y) \|$) between two neighboring pixels, the larger λ is. A larger λ will encourage the smoothness constraint more.
2. U_λ is to normalize $\| I_t(x) - I_t(y) \|$ so that λ should be within a certain range depending on w_s.

[8] Bundle Optimization

Problem :

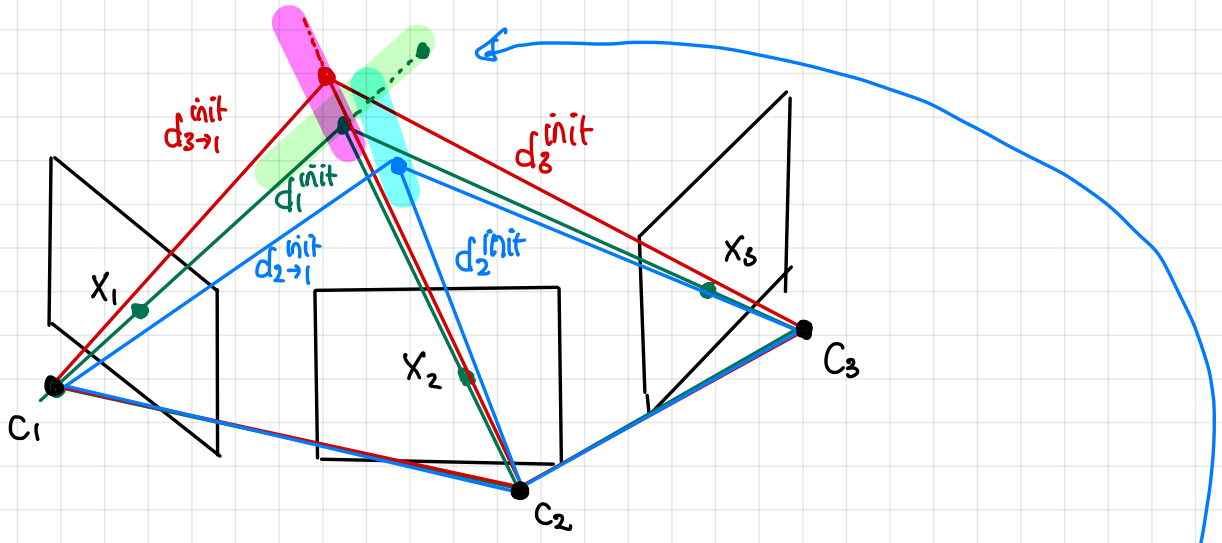


Ideally : $d_1^{init} = d_{2\to1}^{init} = d_{3\to1}^{init}$

Reality : $d_1^{init} \neq d_{2\to1}^{init} \neq d_{3\to1}^{init}$ → This is the cause of the flickering problem

To address the problem we need to make :

$d_1 \approx d_{2\to1} \approx d_{3\to1}$ → Enforce them to be as close as possible.



Basic idea : To find other disparity value other than d_t^{init} that close the gap in d_1 & $d_{2\to1}$ & $d_{3\to1}$

Candidates for d_t^{init} are : $\{d_t^{init} - N\varepsilon, \ldots, d_t^{init}, d_t^{init} + \varepsilon, \ldots, d_t^{init} + N\varepsilon\}$

N & ε are variables that need to be decided.

- Input: $\{I_t, P_t, D_t^{init}\}_{t=1}^{N}$

  Output: D_t

$$D_t^{init} = \underset{\{d_t^{init} + n\varepsilon\}_{n=-N}^{+N}}{\operatorname{argmin}} \sum_x \left[1 - U(x) f_d^{init}(x, D_t(x)) - \sum_{y \in N_x} \lambda(x,y) f_p(D_t(x), D_t(y)) \right]$$

candidates: $\{d_{t-N\varepsilon}^{init}, \ldots, d_{t-\varepsilon}^{init}, d_t^{init}, d_{t+\varepsilon}^{init}, \ldots, d_{t+N\varepsilon}^{init}\}$

Where:

$$f_d(x, d) = \sum_{t'}^{N_t} p_c(x, d, I_t, I_{t'}) \, p_v(x, d, D_{t'})$$

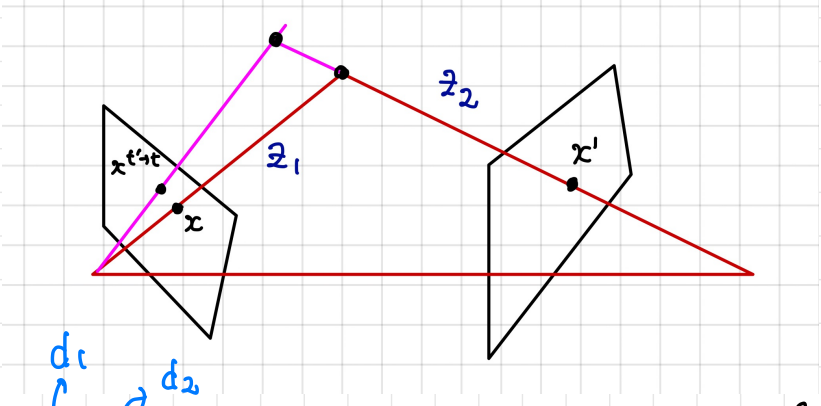
photo consistency — geometric coherence

⇓

the same as in the previous step.

Geometric Coherence:

Goal: to measure how consistent z_1 (or d_1) and z_2 (or d_2) geometrically.



$$p_v(x, d, D_{t'}) = \exp\left(- \frac{\| x - l_{t',t}(x', D_{t'}(x)) \|^2}{2\sigma_d^2} \right)$$

similar to σ_c: to control how tolerant we are with the error

↑

; σ_d = standard deviation

where: $x' = l_{t,t'}(x, d)$

$$x^{t'\to t} = l_{t',t}(x', D_{t'}(x))$$

$$\| x - x^{t'\to t} \| = \| x - l_{t',t}(x', D_{t'}(x)) \|$$