Stereo Reconstruction of Building Interiors with a Vertical Structure Prior

Bernhard Zeisl   Christopher Zach   Marc Pollefeys

Computer Vision and Geometry Group, Institute for Visual Computing, ETH Zurich

Summary of our Work

- For building interiors the open space is bounded by
  (1) parallel ground and ceiling planes,
  (2) vertical wall elements.
- We employ this assumption as a strong prior in dense depth map estimation from stereo images.
- A Dynamic Programming (DP) framework allows to introduce smoothness between vertical elements.
- Besides the reconstruction of vertical structures, the algorithm detects non-vertical regions and allows to fill in plausible extensions.

1 Motivation

Challenges for indoor depth map estimation: weakly textured areas and view dependent highlights.

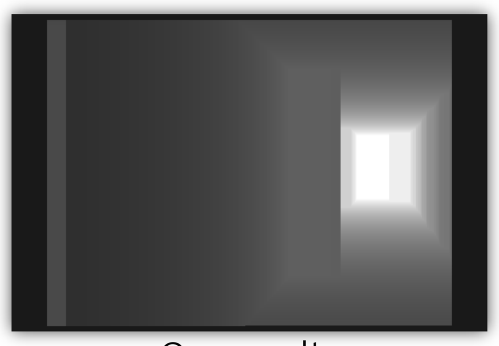

Left image


Right image


Best cost depth

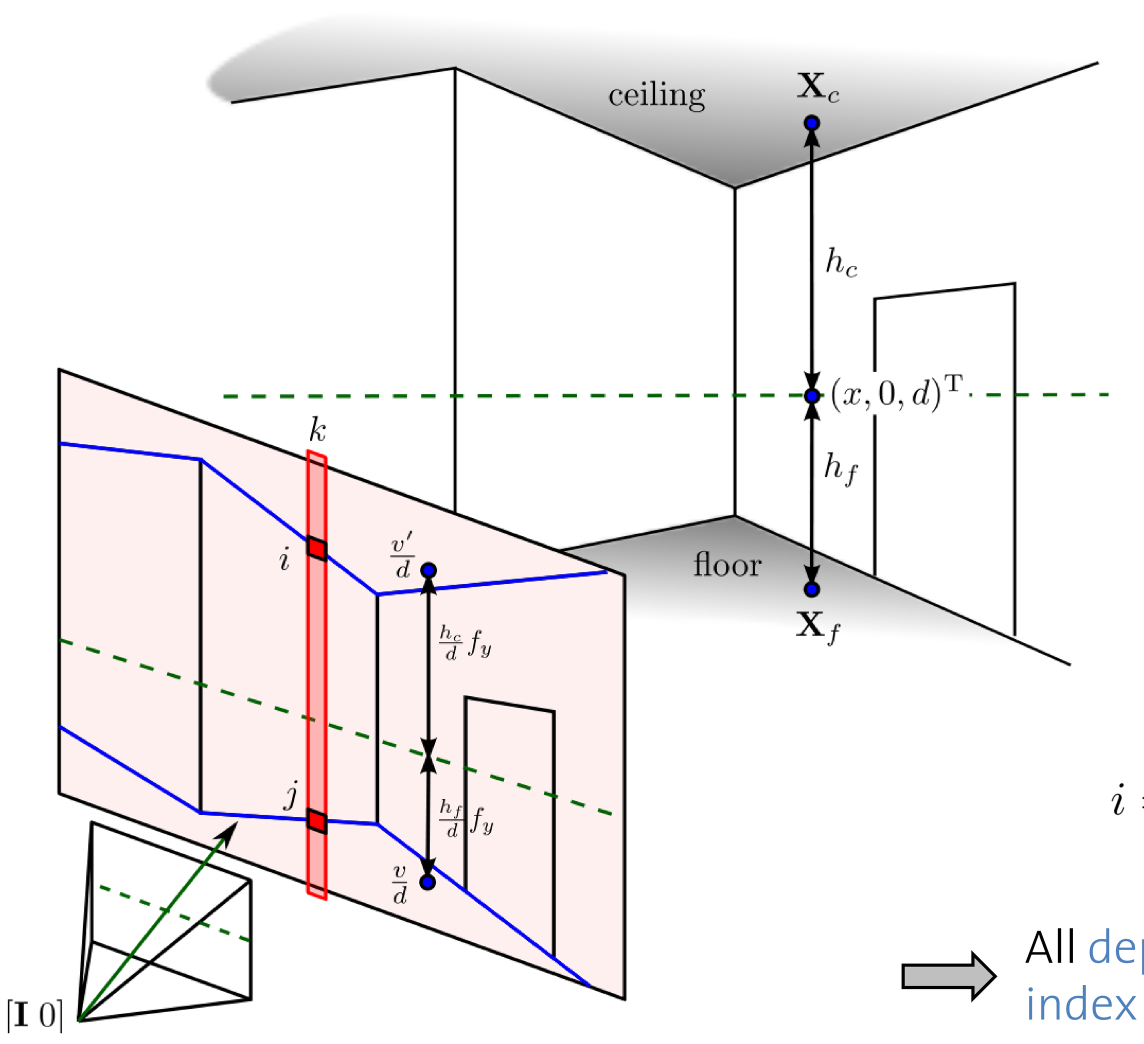

Global optimization (GC)


Scanline optimized


Our result

⇨ Explicit incorporation of vertical world assumption significantly stabilizes the depth map estimation.

2 Vertical Structures



$$\mathbf{X}_f = (x, h_f, d)^T$$
$$\mathbf{X}_c = (x, h_c, d)^T$$
$$(u, v, d)^{\mathrm{T}} = \mathbf{K}\mathbf{X}_f$$
$$(u', v', d)^{\mathrm{T}} = \mathbf{K}\mathbf{X}_c$$
$$\mathbf{K} = \begin{pmatrix} f_x & 0 & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{pmatrix}$$

$$i = \frac{v'}{d} \quad \text{and} \quad j = \frac{v}{d}$$

$$i = \frac{h_c}{h_f} \cdot j + \left(1 - \frac{h_c}{h_f}\right) p_y$$

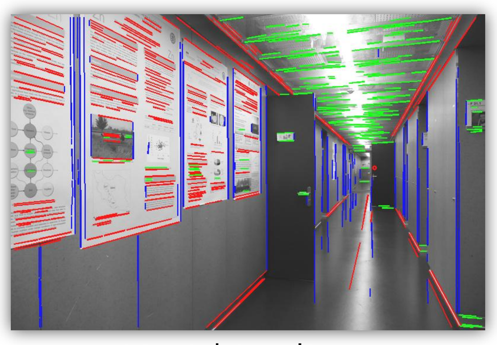
⇨ All depth hypotheses relate to index pairs (i,j): $d \mapsto (i, j)$

3 Preprocessing

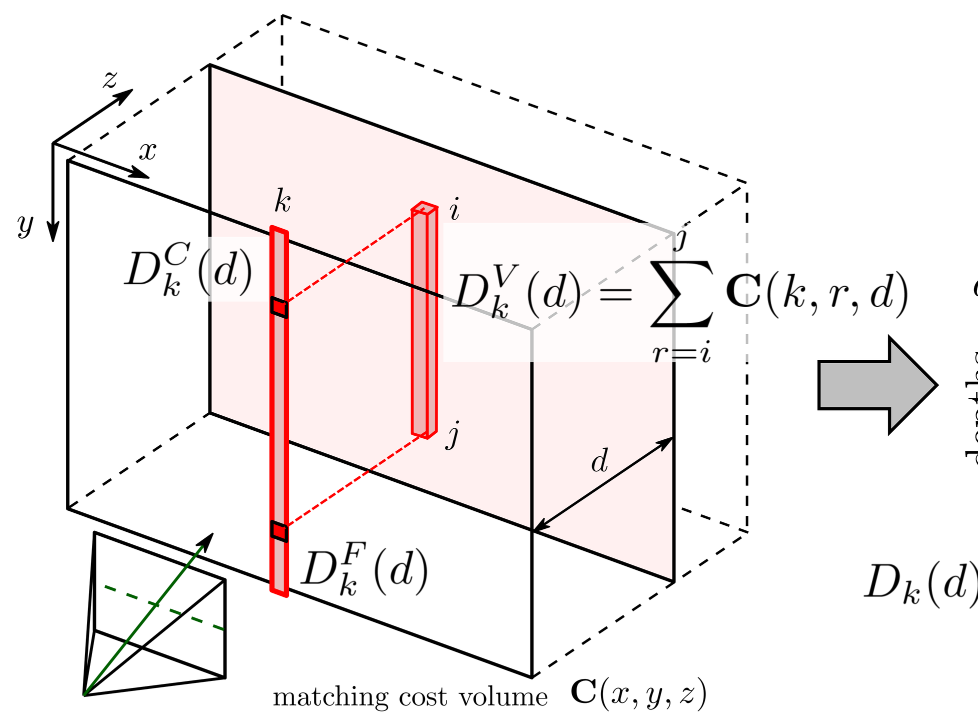
1) **Image alignment with vertical direction**:
3D camera rotation (image warping) moving the vertical vanishing point towards infinity.

3) **Stereo image matching**: Plane sweep approach (along z and y axis), which preserves the previous image alignment.

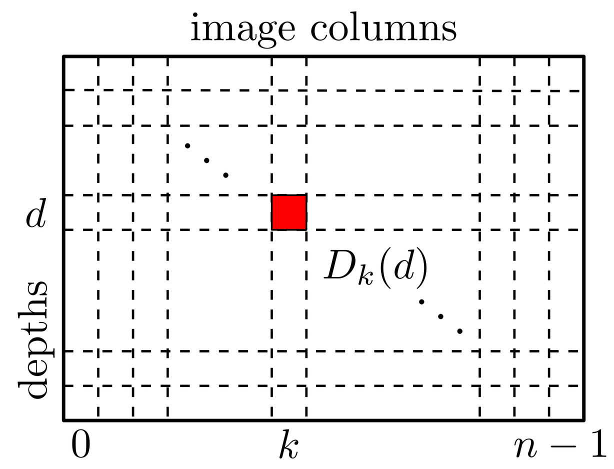
2) **Identification of floor and ceiling plane**:
Robust voting for points on boundary edges.


Vanishing lines

4 Cost Aggregation



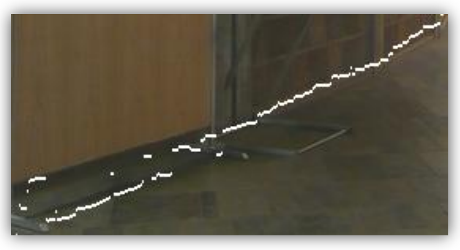
New cost table:



$$D_k(d) = D_k^C(d) + D_k^V(d) + D_k^F(d)$$

Best cost solution:

$$d_k^* = \arg\min_d D_k(d) \quad \forall k \in \{0, \ldots, n-1\}$$


Best cost boundary

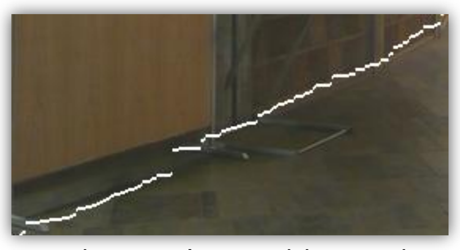
5 Dynamic Programming

$$E = \underbrace{D_0(l_0)}_{} + \sum_{k=1}^{n-1} \{\underbrace{D_k(l_k)}_{} + \underbrace{V(l_k, l_{k-1})}_{}\}$$

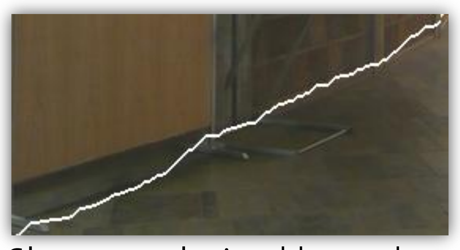
Cost for a vertical structure at column k   Smoothness between neighboring columns

1) **Penalty for depth changes:**
Linear cost, truncated to allow for large changes

$$V(d_k, d_{k-1}) = \lambda_d \min\left(|d_k - d_{k-1}|, t\right)$$


Depth regularized boundary

2) **Extension by slope based smoothness:**
- Labels are binary cliques (depth and slope)
- Small number of slopes sufficient (e.g. 3 or 5)


Slope regularized boundary

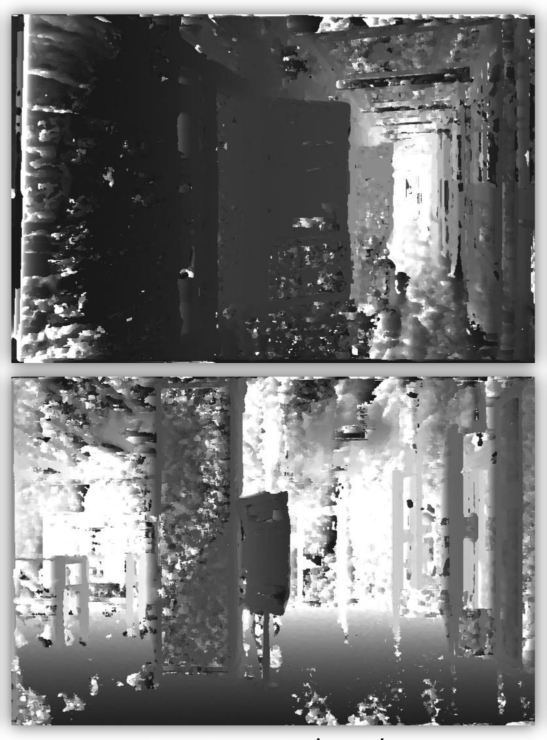
$$V(l_k, l_{k-1}) = \underbrace{\lambda_s |s_k - s_{k-1}|}_{\text{slope penalty}} + \underbrace{\lambda_d |d_k - d_{k-1} - s_{k-1}|}_{\text{depth penalty (compensated by slope)}}$$

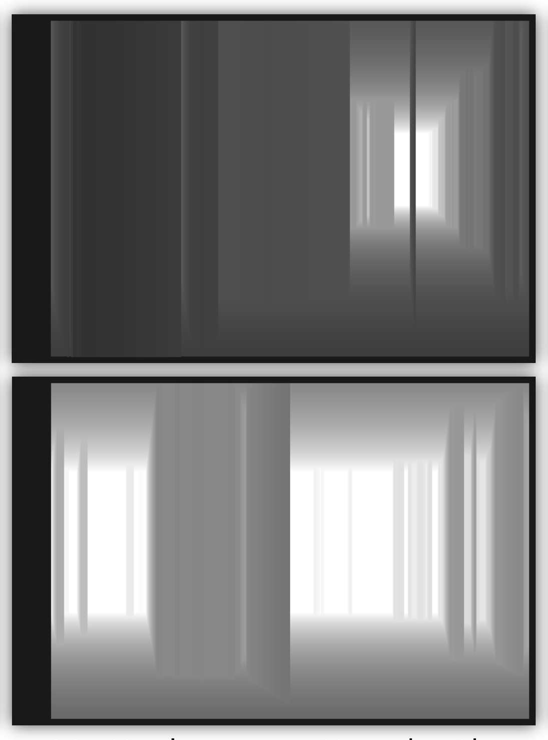
3) **Detection of non-vertical structures**
Model selection via new label for non-vertical structures

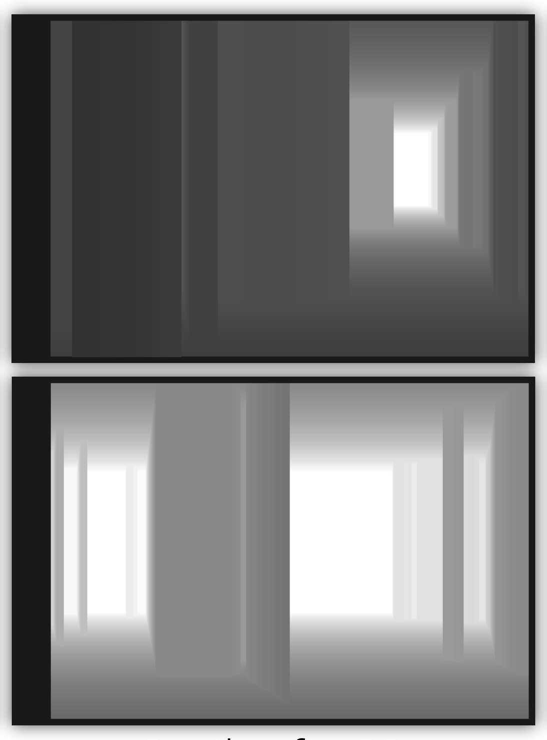
$$D_k(l = \text{non-vertical}) = B + \sum_{r=0}^{m-1} \min_d \mathbf{C}(k, r, (\mathbf{e}_z\ d))$$

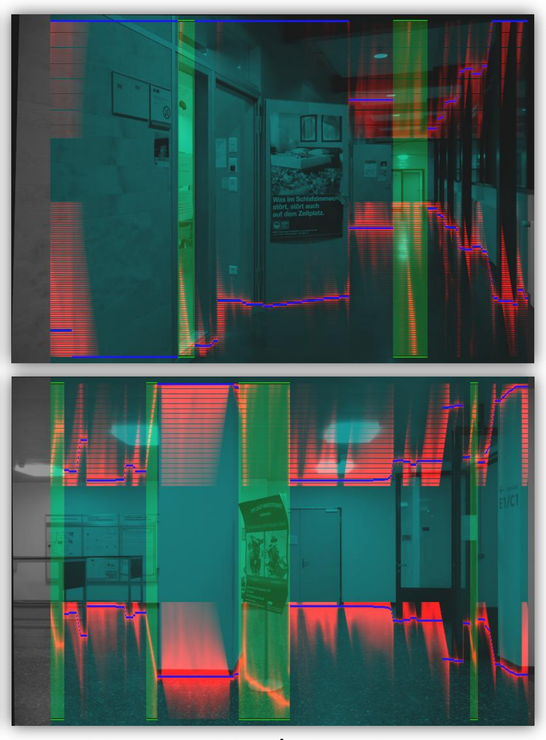
6 Results


Upright image


Best cost depth


Vertical aggregation depth


Depths after DP


Non-vertical structures

Speed Analysis	
Plane sweep (GPU):	160ms
Vertical structure cost aggregation (CPU):	50ms
Dynamic Programming (CPU) for 1 / 3 / 5 slopes:	5 / 46 / 120ms
	215 – 330ms

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

CVG Computer Vision and Geometry Lab

http://cvg.ethz.ch