A dynamic programming algorithm for perceptually-consistent stereo

Jialiang Wang
Daniel Glasner
and
Todd Zickler

TR-02-17

A dynamic programming algorithm for perceptually-consistent stereo

Jialiang Wang
Harvard University
jialiangwang@g.harvard.edu

Daniel Glasner
AiCure *
dglasner@gmail.com

Todd Zickler
Harvard University
zickler@seas.harvard.edu

1. Introduction

This document provides details of the dynamic programming algorithm discussed in *"Towards perceptually-consistent stereo: a scanline study"* [3]. For the motivation of the algorithm, see that paper.

2. Objective

On a horizontal scanline of a rectified pair of stereo images, we assume a discrete domain for the visual field, indexed by $n \in \{1 \ldots N\}$. For notational convenience, we choose a flipped coordinate system, with $n = 1$ the rightmost pixel and $n = N$ the left-most one. We define a small set of spatial basis functions $\{B_b(n)\}_{b=1\ldots M}$ and the disparity at pixel n as $d_\ell(n) = \sum_{b=1}^{M} \ell(b)B_b(n)$ with shape coefficients $\ell = \{\ell(b)\}_{b=1\ldots M}$. The M-dimensional shape space is quantified, so there are L possible choices of vector ℓ. Assume we have matching cost $C(n, \ell) \in [0, 1]$ defined at all (n, ℓ). Suppose $\{s_i\}$ are breakpoint locations between 1 and N, and with $s_{i-1} < s_i$. We define a half-occlusion function $k(s_i, \ell_{s_i}, \ell_{s_{i+1}})$ that dictates the spatial location of a half-occlusion boundary according to the following rule (see the blue dashed line in Figure 1 for an example):

$$
k(s_i, \ell_{s_i}, \ell_{s_{i+1}}) = \\
\begin{cases}
s_i + d_{\ell_{s_i}}(s_i) - d_{\ell_{s_{i+1}}}(s_i), & \text{if } d_{\ell_{s_{i+1}}}(s_i) < d_{\ell_{s_i}}(s_i) \\
s_i, & \text{otherwise.}
\end{cases}
$$

We want to simultaneously find the number of breakpoints, the breakpoint locations $\{s_i\}$, and the per-segment shape coefficients $\{\ell_{s_i}\}$ between each sequential pair of breakpoints. Mathematically, we want to find the global minimum of the objective function:

$$
\sum_i \rho\left(s_{i-1}, s_i, \ell_{s_{i-1}}, \ell_{s_i}\right) = \\
\sum_i \left(\rho_c(s_{i-1}, s_i, \ell_{s_{i-1}}, \ell_{s_i}) + \lambda_1 \rho_g(s_{i-1}, \ell_{s_{i-1}}, \ell_{s_i}) + \lambda_2\right)
\tag{1}
$$

---

*Most work was done when Daniel Glasner was at Harvard University

where $\rho_c(s_{i-1}, s_i, \ell_{s_{i-1}}, \ell_{s_i})$ is the matching cost of segment $[s_{i-1}, s_i)$, $\rho_g(s_{i-1}, \ell_{s_{i-1}}, \ell_{s_i})$ is a boundary cost defined at breakpoint s_{i-1}, and λ_1, λ_2 are constants.

The matching cost of a segment is the sum of the matching cost over the unoccluded portion of the segment:

$$
\rho_c(s_{i-1}, s_i, \ell_{s_{i-1}}, \ell_{s_i}) = \sum_{n=k(s_{i-1}, \ell_{s_{i-1}}, \ell_{s_i})}^{s_i - 1} C\left(n, \ell_{s_i}\right).
$$

3. Optimization by dynamic programming

Let $opt(u, \ell_u)$ denote the scalar cost associated with the optimal disparity sub-profile over interval $[1, u]$ with the constraint that the final segment, the one containing u, has shape ℓ_u. For the first pixel we have $opt(1, \ell) = C(1, \ell) + \lambda_2$ for all ℓ, and from there, we can visit the remaining pixels $u \in \{2 \ldots N\}$ in sequence, recursively computing the L values of $opt(u, \cdot)$ at each pixel u. For each pair (u, ℓ_u), we search for the optimal location of the *previous* breakpoint v. We set $v = 1$ if having only one segment is optimal for interval $[1, u]$. If $v \neq 1$, v will necessarily be a breakpoint between the shape of the final segment ℓ_u and a different shape, say ℓ_v, of the segment before it (see Figure 1). Thus, we can write the recursion as

$$
opt(u, \ell_u) = \min_{\Gamma} \left(\rho(v, u, \ell_v, \ell_u) + opt(v, \ell_v)\right),
\tag{2}
$$

where Γ is a subset of pairs (v, ℓ_v) within $[1, u-1] \times [1 \ldots L]$ that satisfy several constraints (to be discussed in Section 4). To be able to build each valid subset Γ during recursion, we also maintain a record of the optimal beginnings (the minimizers of Equation 2):

$$
arg(u, \ell_u) = \arg\min_{\Gamma} \left(\rho(v, u, \ell_v, \ell_u) + opt(v, \ell_v)\right).
\tag{3}
$$

This data structure has size $N \times L \times 2$. In our notation, $(v, \ell_v) = arg(u, \ell_u)$ means that among all possible sub-profiles defined on interval $[1, u]$ with final segment shape ℓ_u, the one with the lowest cost (and cost equal to $opt(u, \ell_u)$) is smooth over interval $[v, u]$ and has a breakpoint at v marking a transition to shape ℓ_v.
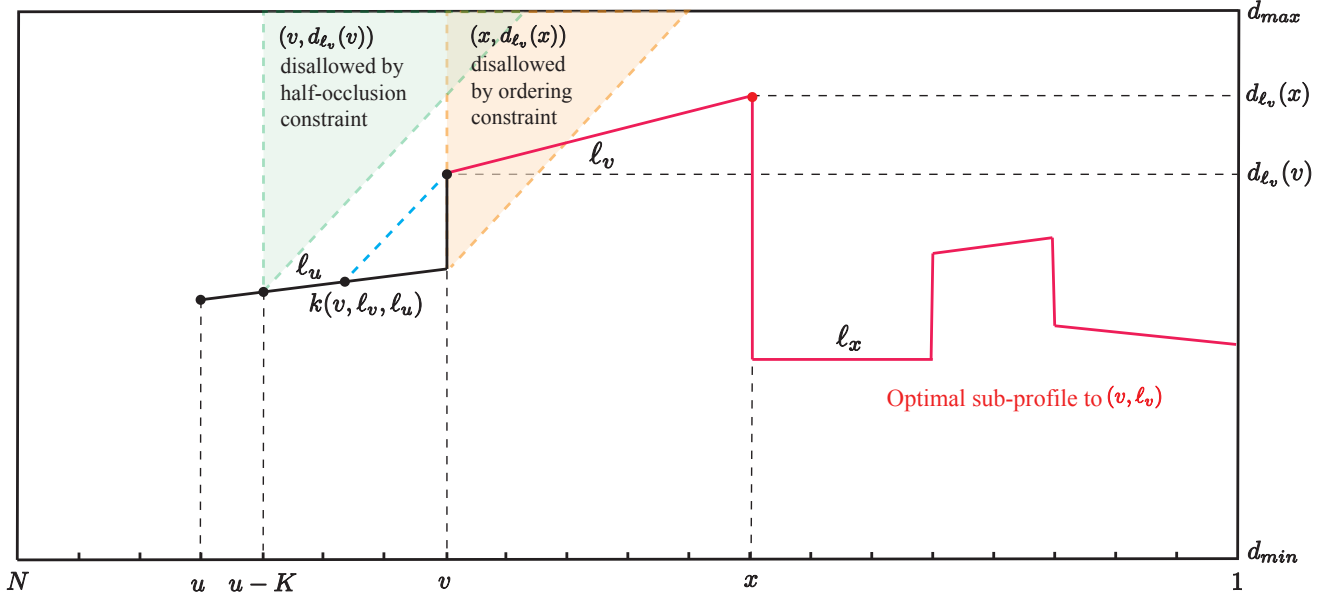
Figure 1. Dynamic programming algorithm and its constraints. At each (u, ℓ_u), we recursively find the optimal location of the previous breakpoint v and its associated model label ℓ_v by making use of the previously-computed sub-profiles for all pixels $v < u$. Some (v, ℓ_v) combinations are excluded from consideration by imposing several constraints. First, $(v, d_{\ell_v}(v))$ cannot be in the green region per half-occlusion constraint. Second, to ensure geometric ordering, we disallow (v, ℓ_v) for which the previously-computed optimal sub-profile to (v, ℓ_v) (red line) has a leftmost breakpoint x such that $(x, d_{\ell_v}(x))$ is in the orange region (Eqn. 4).

Algorithm 1 Find optimal disparity profile (from [3])

1: **for all** $\ell \in [1, L]$ **do**
2: $\quad opt(1, \ell) \leftarrow C(1, \ell) + \lambda_2$
3: $\quad arg(1, \ell) \leftarrow (1, \ell)$
4: **end for**
5: **for** $u \leftarrow$ *2 to N* **do**
6: $\quad$ **for** $\ell_u \leftarrow$ *1 to L* **do**
7: $\quad\quad \Gamma \leftarrow \text{BUILDVALIDSUBSET}(u, \ell_u, arg(u, \ell_u))$
8: $\quad\quad opt(u, \ell_u) \leftarrow \min_{\Gamma} \left(\rho(v, u, \ell_v, \ell_u) + opt(v, \ell_v) \right)$
9: $\quad\quad arg(u, \ell_u) \leftarrow \arg\min_{\Gamma} \left(\rho(v, u, \ell_v, \ell_u) + opt(v, \ell_v) \right)$
10: $\quad$ **end for**
11: **end for**
12: $\mathcal{I}, u \leftarrow N$ $\qquad\qquad\qquad$ ▷ initialize trace back
13: $\Theta, \ell_u \leftarrow \arg\min_{\ell} opt(N, \ell)$
14: **while** $u > 1$ **do**
15: $\quad (u, \ell_u) \leftarrow arg(u, \ell_u)$
16: $\quad \text{APPEND}(\mathcal{I}, u)$
17: $\quad \text{APPEND}(\Theta, \ell_u)$
18: **end while**

Once the recursion terminates at pixel N, we trace the optimal profile by using the arg data structure to accumulate the profile's breakpoints and shape-transitions, from the last one at pixel N to the first one at pixel 1. Algorithm 1 provides pseudo code (from [3]).

4. Constraints

To discuss the constraints, we first need to define two types of breakpoints: We call a breakpoint an *occluding*

breakpoint if $d_{\ell_u}(v) < d_{\ell_v}(v)$; otherwise we call it a *non-occluding* one. In Figure 1, the breakpoint at v is an occluding one while the one at x is not.

There are three constraints we enforce in our algorithm. We first introduce all three constraints in Section 4.1, and then discuss the limitations of our "ordering constraint" in Section 4.2.

4.1. Description of constraints

Disparity bounds. We specify a valid disparity range $[d_{min}, d_{max}]$, such that $d_{min} < d_\ell(n) < d_{max}$. Following this constraint, some models l are only valid at a subset of the pixel locations.

Half-occlusion constraint. We enforce a half-occlusion constraint which imposes a hard lower bound K on the size of the visible portion of segments that include half-occlusion. This type of segment occurs after all occluding breakpoints. Mathematically, this constraint is $u - k(v, \ell_v, \ell_u) \geq K$ if $d_{\ell_u}(v) < d_{\ell_v}(v)$. For example, the green triangular region in Figure 1 is excluded from Γ.

Ordering constraint. As in most dynamic programming approaches to scanline stereo, our algorithm is only made feasible when the output preserves geometric ordering [1], which, as shown in Figure 2, means that the ordering of matched points is preserved between the left and the right images. In our algorithm, we enforce this with a constraint that is sufficient but not necessary. At an occluding breakpoint, we require:
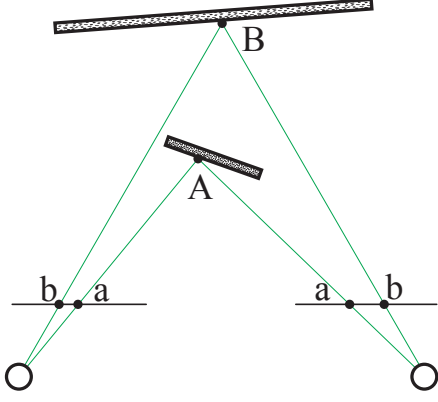
Figure 2. An example where the geometric ordering constraint is violated. Points AB are projected as ba in the left camera but ab in the right camera.

$$v - x > d_{\ell_v}(x) - d_{\ell_u}(v) \quad \text{with } (x, \ell_x) = arg(v, \ell_v) \quad (4)$$

This constraint requires that the occluding segment, *e.g.* segment $[x, v]$ in Figure 1, is long enough to guarantee that geometric ordering is preserved. It can be visualized as a requirement on the second breakpoint x from u, and the disparity $d_{\ell_v}(x)$ at that point. For example, any (v, ℓ_v) pair whose previous optimal breakpoint (stored in arg) falls into the orange region in Figure 1 is excluded from Γ.

Imposing this constraint is sufficient to guarantee geometric ordering. Figure 3 shows the depth planes corresponding to the disparity segments with shape coefficients ℓ_u and ℓ_v from Figure 1. V and X are the back projec-
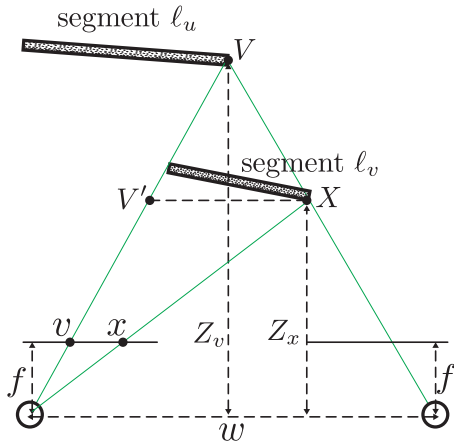

Figure 3. The front plane (segment ℓ_v) causes half-occlusion in the background plane (segment ℓ_u). The ordering constraint is preserved if point V is not visible from the right camera. This imposes a bound (Eq. 4) on the minimum projected length of the front plane in the left camera (*i.e.* $|xv|$).

tion of breakpoints v and x in the left camera. Let $d_{\ell_u}(v)$, $d_{\ell_v}(x)$ be the disparities of background pixel v and foreground pixel x respectively. We know that [2],

$$d_{\ell_u}(v) = f\frac{w}{Z_v} \quad \text{and} \quad d_{\ell_v}(x) = f\frac{w}{Z_x}.$$

for focal length f and baseline w.

If point V in the scene is not visible from the right camera, then geometric ordering is guaranteed. We define V' as the intersection of the ray through V and a line through X that is parallel to baseline w. From similar triangles,

$$\frac{|V'X|}{w} > \frac{Z_v - Z_x}{Z_v}$$

$$\Leftrightarrow \quad |V'X| > w\left(1 - \frac{Z_x}{Z_v}\right) = w\left(1 - \frac{d_{\ell_u}(v)}{d_{\ell_v}(x)}\right)$$

Using similar triangles again, we have,

$$\frac{v - x}{f} = \frac{|V'X|}{Z_x}$$

$$\Leftrightarrow \quad v - x = |V'X|\frac{f}{Z_x} > w\left(1 - \frac{d_{\ell_u}(v)}{d_{\ell_v}(x)}\right)\frac{d_{\ell_v}(x)}{w}$$

$$> d_{\ell_v}(x) - d_{\ell_u}(v),$$

of which is exactly the constraint of Equation 4.

4.2. Limitations of the ordering constraint

Our ordering constraint is sufficient but not necessary to garantee geometric ordering. This means that there are some profiles that do not violate geometric ordering but could not be reconstructed by our algorithm. Figure 4 is an example. Figure 4(a) shows the scene geometry and 4(b) shows the disparity profile. Our constraint excludes this due to the short segments $[x, v]$, even though the true scene geometry does not violate geometric ordering.

Another example is Figure 5, which results from the fact that the optimization we do is over the previous breakpoint v instead of a joint optimization over the two previous breakpoints. In this case, $arg(v, \ell_v)$ stores the black profile to (v, ℓ_v) since it has the lowest cost, and the blue one is not stored. Thus, at (u, ℓ_u), the black profile is excluded from Γ because it violates the ordering constraint. As a result, the algorithm selects the red profile which has a different previous breakpoint (v', ℓ_v') instead of the blue one, although the blue one has lower cost and preserves geometric ordering.

ference on Artificial intelligence-Volume 2, pages 631–636. Morgan Kaufmann Publishers Inc., 1981.

[2] P. N. Belhumeur. A bayesian approach to binocular steropsis. *International Journal of Computer Vision*, 19(3):237–260, 1996.

[3] J. Wang, D. Glasner, and T. Zickler. Toward perceptually-consistent stereo: A scanline study. *Proc. International Conference on Computer Vision (ICCV)*, 2017.
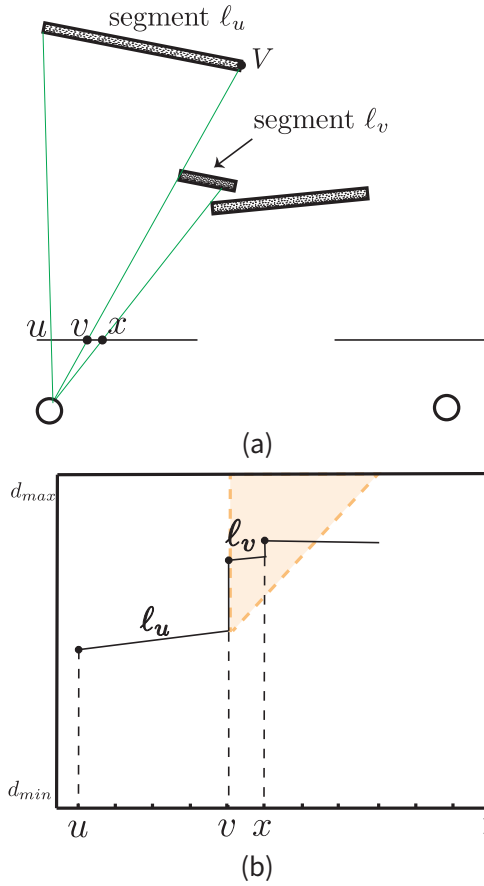
Figure 4. An example that cannot be recovered by our algorithm even though it does not violate the geometric ordering. (a) the depth scene and (b) the corresponding disparity profile.
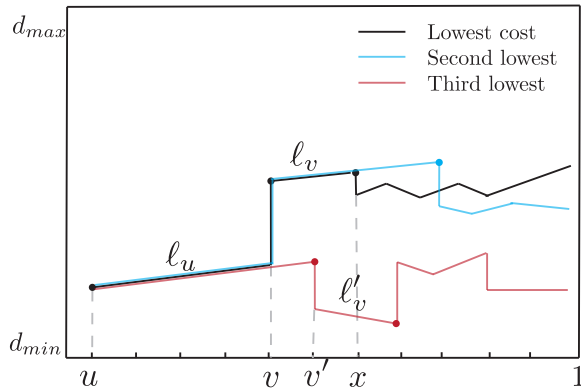

Figure 5. An example of the limitations of the ordering constraint. The red profile is selected instead of the blue one, even though neither one violates geometric ordering, and the blue one has lower cost.

References

[1] H. H. Baker and T. O. Binford. Depth from edge and intensity based stereo. In *Proceedings of the 7th international joint con-*