Main Idea: If we knew the surface slants, we could replace fronto-parallel bias with bias parallel to surface. But, how would we do it in a discrete optimization setting? The solution is to rasterize a disparity surface at an arbitrary depth and adjust $V(d, d')$ to follow the discrete steps.

2D Orientation Prior:

$$V_S(d_p, d'_p) = V(d_p + f_p(d_p), d'_p)$$

offset depends on pixel location

$$f_p = (-1, 0, +1)$$

3D Orientation Prior:

$$V_S(d_p, d'_p) = V(d_p + f_p(d_p), d'_p)$$

offset depends on pixel location and disparity

Surface Normal Priors

- We derive the exact relationship between surface normals in scene coordinates and disparity surface orientations
- Requires 3D orientation prior; disparity steps no longer aligned vertically

Where do we get priors?

- Matched features + triangulation
- Matched features + plane fitting
- Low-res matching + plane fitting
- Semantic prediction
- Manhattan-world assumptions
- GT oracles

Contributions

- SGM’s fronto parallel surface assumption makes it fail on untextured slanted surfaces
- Our extension (SGM-P) utilizes precomputed surface orientation priors to reduce such errors

Advantages

- Minimal runtime overhead
- Easy to add to existing SGM implementation

Experiments

- SGM vs SGM-P (various forms of prior)
- Improvements for slanted textureless surfaces; never hurts performance

Summary

- Extension of SGM: uses precomputed surface orientation priors
- Huge performance gains for slanted textureless scenes
- Soft constraint, doesn’t hurt performance; small overhead
- Explore predicting more accurate orientation priors in future.