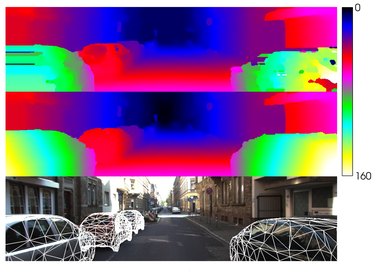
Displets: Resolving Stereo Ambiguities using Object Knowledge

Fatma Güney, Andreas Geiger
MPI for Intelligent Systems, Tübingen



Motivation. Using object knowledge, we encourage disparities to agree with plausible surfaces while simultaneously recovering 3D geometry of the objects (bottom). This improves results (center) compared to current state-of-the-art stereo methods (top) [1].

Abstract

- Current stereo methods often fail at textureless, reflecting or semi-transparent surfaces such as cars.
- Yet, as humans we are able to effortlessly extract information about the geometry of objects even from a single image.
- Little is known about the importance of recognition for stereo matching.

We introduce object knowledge for well-constrained object categories into a slanted-plane MRF. We leverage inverse graphics to sample set of plausible object disparity maps given an initial semi-dense disparity estimate and a rough semantic segmentation of the image. We encourage the presence of these 2.5D shape samples (or "displets") depending on how much their geometry and semantic class agrees with the image observations.

Representation

We decompose the image into planar superpixels $\mathcal{S}$. Each **superpixel** $i \in \mathcal{S}$ is associated with:

- region $\mathcal{R}_i$ in the image
- random variable $\mathbf{n}_i \in \mathbb{R}^3$ describing a plane in 3D

Each **displet** $k \in \mathcal{D}$ is associated with:

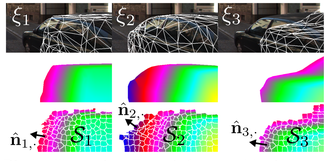
- random variable $d_k \in \{0, 1\}$ (presence/absence)

Our goal is to **infer**:

- superpixel planes $\mathbf{n} = \{\mathbf{n}_i | i \in \mathcal{S}\}$
- displets $\mathbf{d} = \{d_k | k \in \mathcal{D}\}$

Displets

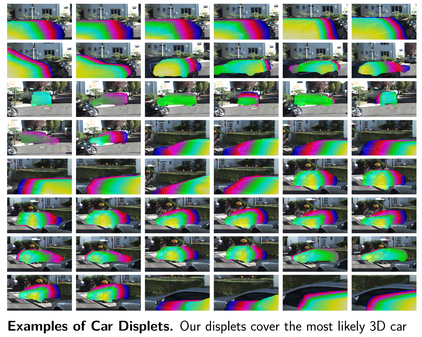
Displets: A representative set of disparity maps for a specific semantic class (e.g., car) conditioned on the image.



Displet Illustration. Top: wireframe model, middle: rendered disparity map, bottom: fitted superpixel planes.

We sample 3D CAD model configurations ξ_k and associate each displet $k \in \mathcal{D}$ with

- its class label c_k
- a fitness score κ_k
- a set of superpixels $\mathcal{S}_k$
- the corresponding planes $\hat{\mathbf{n}}_{k,1}, \ldots, \hat{\mathbf{n}}_{k,|\mathcal{S}_k|}$



Examples of Car Displets. Our displets cover the most likely 3D car shapes given the initial disparity map and a semantic segmentation.

Stereo Matching using Displets

$$E(\mathbf{n}, \mathbf{d}) = \underbrace{\sum_{i \in \mathcal{S}} \varphi_i^{\mathcal{S}}(\mathbf{n}_i)}_{\text{Data}} + \underbrace{\sum_{i \sim j} \psi_{ij}^{\mathcal{S}}(\mathbf{n}_i, \mathbf{n}_j)}_{\text{Smoothness}} + \underbrace{\sum_{k \in \mathcal{D}} \varphi_k^{\mathcal{D}}(d_k)}_{\text{Displets}} + \underbrace{\sum_{k \in \mathcal{D}} \sum_{i \in \mathcal{S}_k} \psi_{ki}^{\mathcal{D}}(d_k, \mathbf{n}_i)}_{\text{Consistency}}$$

Data term

We penalize deviations from an initial sparse disparity map $\hat{\Omega}$ obtained using the method of [1]:

$$\varphi_i^{\mathcal{S}}(\mathbf{n}_i) = \sum_{\mathbf{p} \in \mathcal{R}_i \cap \hat{\Omega}_+} \rho_{\tau_1}(\omega(\mathbf{n}_i, \mathbf{p}) - \hat{\omega}(\mathbf{p}))$$

- $\omega(\mathbf{n}_i, \mathbf{p})$ denotes the disparity of plane $\mathbf{n}_i$ at pixel $\mathbf{p}$.
- $\rho_\tau(\cdot)$ is a robust l_1 penalty $\rho_\tau(x) = \min(x, \tau)$.

Local smoothness

For adjacent superpixels, we penalize discontinuities at the boundary $\mathcal{B}_{ij}$ and encourage similar orientations:

$$\psi_{ij}^{\mathcal{S}}(\mathbf{n}_i, \mathbf{n}_j) = \theta_1 \sum_{\mathbf{p} \in \mathcal{B}_{ij}} \rho_{\tau_2}(\omega(\mathbf{n}_i, \mathbf{p}) - \omega(\mathbf{n}_j, \mathbf{p})) \; + \; \theta_2 \; \rho_{\tau_3}\left(1 - |\mathbf{n}_i^T \mathbf{n}_j| / (\|\mathbf{n}_i\| \|\mathbf{n}_j\|)\right)$$

Displet Unary

We encourage image regions with semantic class label c_k to be explained by a displet of the corresponding class:

$$\varphi_k^{\mathcal{D}}(d_k) = -\theta_3 [d_k = 1] \cdot (|[\mathbf{S} = c_k] \cap \mathbf{M}_k| + \kappa_k)$$

- $\mathbf{M}_k$ is the foreground mask of displet k.
- $\mathbf{S}$ denotes the semantic segmentation.

Consistency between Displets and Superpixels

We ensure consistency by encouraging superpixels to agree with the geometry of the active displets:

$$\psi_{ki}^{\mathcal{D}}(d_k, \mathbf{n}_i) = \lambda_{ki} [d_k = 1] \cdot (1 - \delta(\mathbf{n}_i, \hat{\mathbf{n}}_{k,z_i}))$$

- λ_{ki} denotes a weight which depends on the distance to the boundary (see experiments).

Inference

Minimizing $E(\mathbf{n}, \mathbf{d})$ is a non-convex, mixed continuous-discrete optimization problem. We use max-product particle BP with TRW-S for the inner iterations.

At each outer iteration, particles are sampled

- around the previous MAP estimate
- using neighbouring superpixels

Sampling Displet Proposals

We leverage the initial semi-dense disparity map, a semantic segmentation of the image as well as 3D CAD models to sub-sample the space of displets using MCMC.

Observation Model

We sample pose parameters from a model which

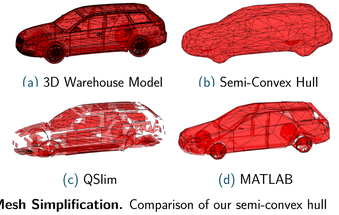
- encourages displets to explain pixels in object regions
- avoids occlusion of other objects

We run one Markov chain for each combination of 3D CAD models and object proposals using Metropolis-within-Gibbs sampling with randomly chosen blocks.

We select the 8 most dominant modes after burn-in for each model and combine all results to yield the final set of displets.

3D CAD Models

For sampling displets, we leverage 3D CAD models from Google Warehouse. To speed-up the rendering process, we reduce details while preserving the overall shape by fitting a "semi-convex hull" to the point clouds of the original models:

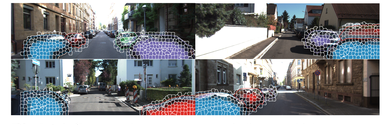


(a) 3D Warehouse Model (b) Semi-Convex Hull (c) QSlim (d) MATLAB

Mesh Simplification. Comparison of our semi-convex hull algorithm to QSlim and MATLAB's `reducepatch` function.

Experimental Results on KITTI

Illustration of Displet Influence λ_{ki}

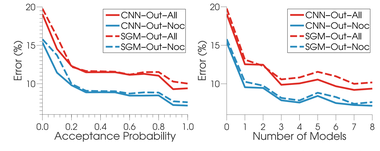


Displet Influence. Less transparency indicates a higher penalty, e.g., we allow more deviation at the displet boundaries.

Ablation Study

	Reflective	All
Input $\hat{\Omega}$ (Interpolated)	19.84 %	3.35 %
Unary Only	17.72 %	3.31 %
Unary + Pair (Boundary)	15.96 %	3.21 %
Unary + Pair (Normal)	17.06 %	3.28 %
Unary + Pair	14.78 %	3.12 %
Unary + Pair + Occ	15.32 %	3.04 %
Unary + Pair + Disp	7.08 %	2.87 %
Unary + Pair + Occ + Disp	7.16 %	2.78 %

Importance of Different Terms.



Number of Proposals and Models. Influence of limiting the number of object proposals (left) and the variety of CAD models used for generating the displets (right).

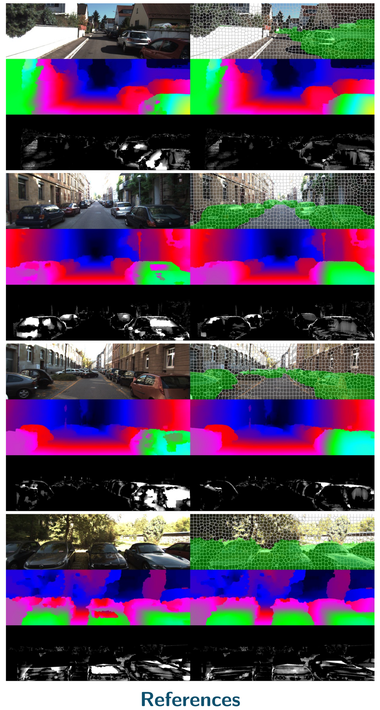
Quantitative Results

Rank	Method	Out-Noc	Out-All	Avg-Noc	Avg-All
1	Our Method	**8.40 %**	**9.89 %**	**1.9 px**	**2.3 px**
2	VC-SF* [2]	11.58 %	12.29 %	2.7 px	2.8 px
3	PCBP-SS [3]	14.26 %	18.33 %	2.4 px	3.9 px
4	SPS-StFl* [4]	14.74 %	18.00 %	2.9 px	3.6 px
,	,	,	,	,	,
11	MC-CNN [1]	18.45 %	21.96 %	3.5 px	4.3 px

Reflective Regions.

Rank	Method	Out-Noc	Out-All	Avg-Noc	Avg-All
1	Our Method	**2.47 %**	**3.27 %**	**0.7 px**	**0.9 px**
2	MC-CNN [1]	2.61 %	3.84 %	0.8 px	1.0 px
3	SPS-StFl* [4]	2.83 %	3.64 %	0.8 px	**0.9 px**
4	VC-SF* [2]	3.05 %	3.31 %	0.8 px	0.8 px
5	SPS-St [4]	3.39 %	4.41 %	0.9 px	1.0 px
6	PCBP-SS [3]	3.40 %	4.72 %	0.8 px	1.0 px

All Regions.



References

[1] Jure Zbontar and Yann LeCun. Computing the stereo matching cost with a convolutional neural network. In *CVPR*, 2015.

[2] Christoph Vogel, Stefan Roth, and Konrad Schindler. View-consistent 3D scene flow estimation over multiple frames. In *ECCV*, 2014.

[3] K. Yamaguchi, D. McAllester, and R. Urtasun. Robust monocular epipolar flow estimation. In *CVPR*, 2013.

[4] Koichiro Yamaguchi, David McAllester, and Raquel Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *ECCV*, 2014.

Contact Information

Web http://www.cvlibs.net/projects/displets
Email {fatma.guney, andreas.geiger}@tue.mpg.de