Learning Sparse High Dimensional Filters: Image Filtering, Dense CRFs and Bilateral Neural Networks

Varun Jampani[1], Martin Kiefel[1,2], Peter V. Gehler[1,2]
[1]MPI for Intelligent Systems, Tübingen; [2]Bernstein Center for Computational Neuroscience, Tübingen

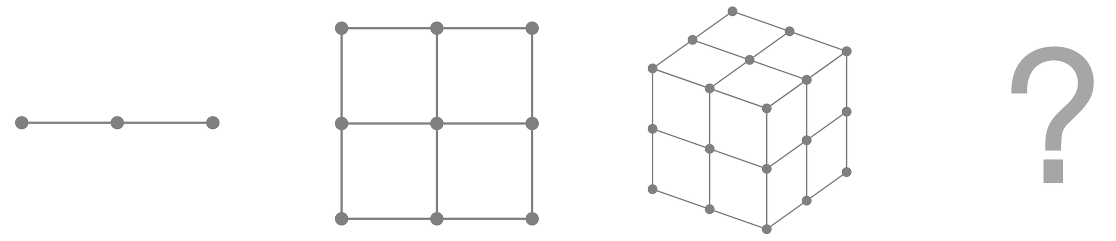
MAX-PLANCK-GESELLSCHAFT
Eberhard Karls Universität Tübingen

We learn image adaptive filters. This generalizes DenseCRF and enables bilateral CNNs.

Code: `http://bilateralnn.is.tuebingen.mpg.de`

1 Sparse High Dimensional Filtering

Problem: Learn filters in high dimensional space



1-D **2-D** **3-D** **N-D**

For example, *Bilateral Filtering* an input $\mathbf{v}$, and features $\mathbf{f}$:

$$\mathbf{v}'_i = \sum_{j \in \mathcal{N}} e^{-\frac{1}{2\sigma^2}\|\mathbf{f}_i - \mathbf{f}_j\|^2} \mathbf{v}_j$$

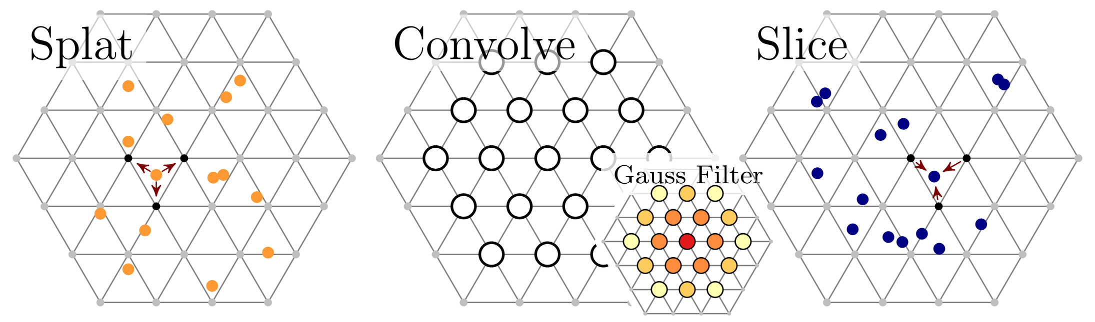
A common choice is pixel position and color: $\mathbf{f} = (x, y, r, g, b)$. Using position features, $\mathbf{f} = (x, y)$ corresponds to standard spatial Gaussian convolution.

Bilateral filters are image dependent | **Bilateral filter output $\mathbf{f} = (x, y, r, g, b)$** | **Spatial Gauss filter output**

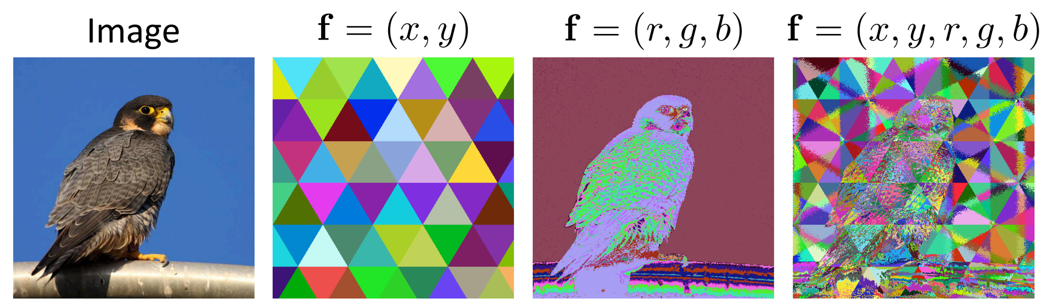
2 Learning Permutohedral Lattice Filters

We use the *permutohedral lattice* from [2].

$$\mathbf{v}' = S_{\text{slice}} W S_{\text{splat}} \mathbf{v}$$



Bilateral filtering using permutohedral lattice



Lattice Visualization. Pixels in the same simplex are shown with the same color

Parameterize the filter kernel instead of using Gaussian. We also generalize to *non-separable* filters.

Learn permutohedral filter via stochastic gradient descent.

Splat, *Convolve* and *Slice* are linear operations and differentiable.

Gradients of a scalar loss L, with respect to –

Input, $\mathbf{v}$:

$$\frac{\partial L}{\partial \mathbf{v}} = S'_{\text{splat}} W' S'_{\text{slice}} \frac{\partial L}{\partial \mathbf{v}'}$$

Filter weights, W:

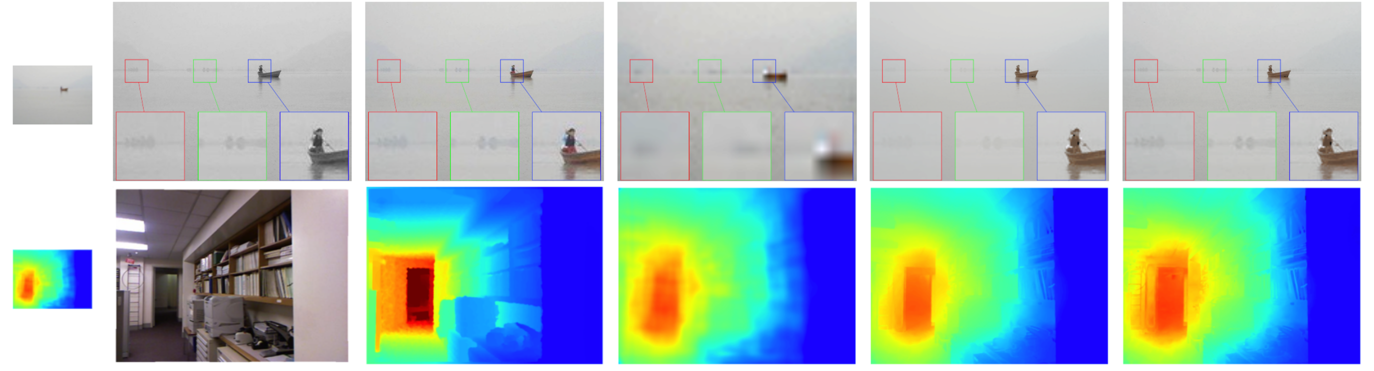
$$\frac{\partial L}{\partial (W)_{i,j}} = \left(S'_{\text{slice}} \frac{\partial L}{\partial \mathbf{v}}\right)_i (S_{\text{splat}} \mathbf{v})_j$$

Consequences:

- *Fast and learnable* high dimensional filtering.
- *Image adaptive* filtering.
- Filtering *unordered* set of points $\{(\mathbf{f}_i, \mathbf{v}_i)\}_{i=1}^n$ (e.g. sparse 3D points).
- Input and output points can be *different*.

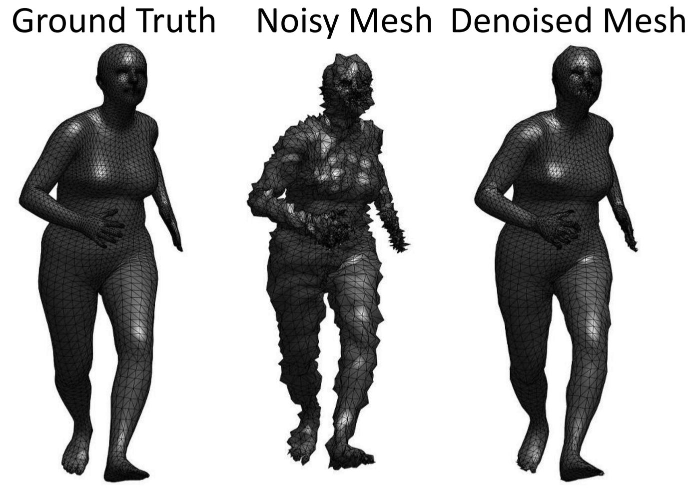
3 Single Filter Applications

1. *Joint Bilateral Upsampling*: Upsample a low-resolution result using a high-resolution guidance image [3].



Sample result of color upsampling (above) and depth upsampling (below)

2. *3D Mesh Denoising*



3D mesh denoising using *learned* bilateral filtering

4 Generalization of DenseCRF

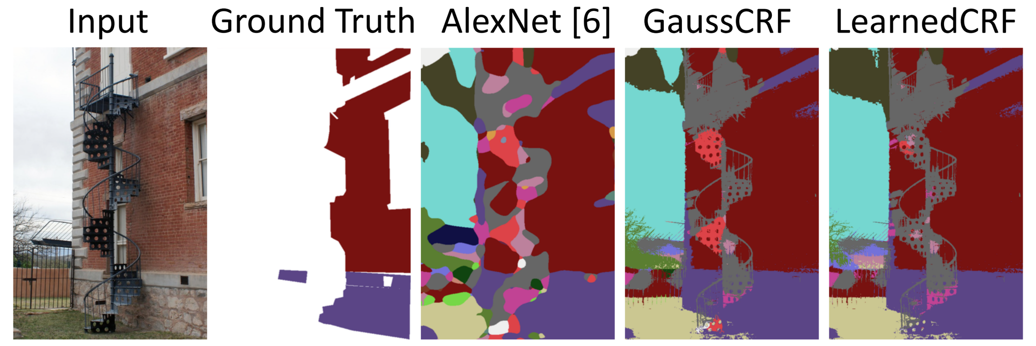
Mean-field inference in densely connected CRFs is tractable [4]. It reduces to bilateral filtering:

$$\underbrace{q_i^{t+1}(x_i)}_{\text{beliefs}} = \frac{1}{Z_i} \exp\{-\underbrace{\psi_u(x_i)}_{\text{unary}} - \sum_{l \in \mathcal{L}} \overbrace{\underbrace{\sum_{j \neq i} \psi_p^{ij}(x_i, l)\, q_j^t(x_j)}_{\text{bilateral filtering}}}^{\text{pairwise}}\}$$

In [4]: Gaussian edge potentials:

$$\psi_p^{ij}(x_i, x_j) = \mu(x_i, x_j) \exp(-(\mathbf{f}_i - \mathbf{f}_j)^\top \Sigma^{-1} (\mathbf{f}_i - \mathbf{f}_j))$$

Here: *Learn pairwise potentials* with back-propagation through truncated message passing.



Sample result of material segmentation

Total/Class accuracy on MINC dataset [6] with AlexNet unaries [6]

	+ MF-1step	+ MF-2 step	+ *loose* MF-2 step
Material segmentation (Pixel Accuracy) - Unary: 67.21 / 69.23			
Gauss CRF	+7.91 / +6.28	+9.68 / +7.35	+9.68 / +7.35
Learned CRF	+9.48 / +6.23	+11.89 / +6.93	**+11.91 / +6.93**

IoU on Pascal VOC12 [5] val./test dataset with DeepLab unaries [7]

	+ MF-1step	+ MF-2 step	+ *loose* MF-2 step
Semantic segmentation (IoU) - Unary: 72.08 / 66.95			
Gauss CRF	+2.48	+3.38	+3.38 / +3.00
Learned CRF	+2.93	+3.71	**+3.85 / +3.37**

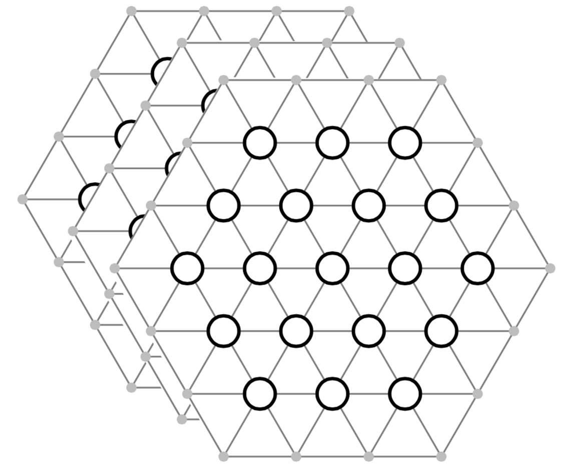
5 Bilateral Neural Networks

Horizontal and vertical *stacking* of high-dimensional filters with *end-to-end* learning.

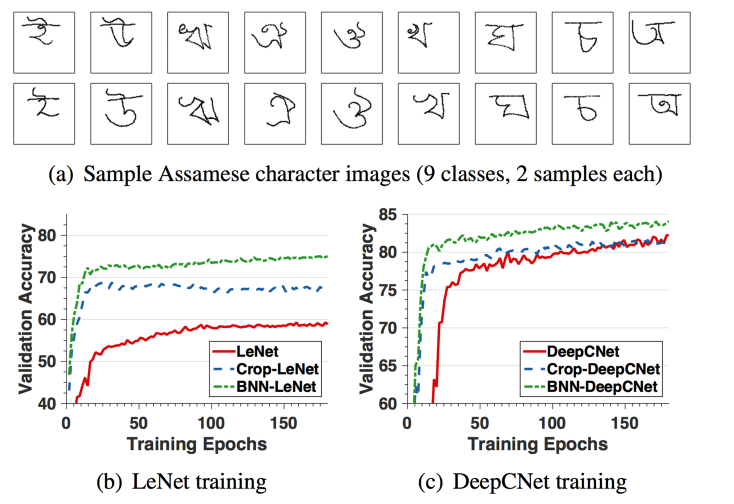
Enables *Bilateral Neural Networks (BNN)*: High dimensional data; unordered (sparse) set of inputs; needs no grid layout.

Example: *Character recognition with hand-written input.*

- Splat and filter only sparse foreground points.



Permutohedral filter bank



Faster and better convergence with BNNs

CPU/GPU runtime (in ms) comparisons, d-dim Caffe vs. Bilateral Convolutional Layer (BCL)

Dim.-Features	d-dim caffe	BCL
2D-(x, y)	**3.3 ± 0.3 / 0.5 ± 0.1**	4.8 ± 0.5 / 2.8 ± 0.4
3D-(r, g, b)	364.5 ± 43.2 / 12.1 ± 0.4	**5.1 ± 0.7 / 3.2 ± 0.4**
4D-(x, r, g, b)	30741.8 ± 9170.9 / 1446.2 ± 304.7	**6.2 ± 0.7 / 3.8 ± 0.5**
5D-(x, y, r, g, b)	out of memory	**7.6 ± 0.4 / 4.5 ± 0.4**

Generalization to non-separable fully parameterized filters results in runtime *linear* in feature dimensions.

6 Conclusion

We propose a filter parameterization and learning technique for sparse and high dimensional data. This can be used for fast learnable bilateral filtering or general high dimensional filtering.

References:

1. Aurich, V., & Weule, J. Non-linear Gaussian filters performing edge preserving diffusion. *In Mustererkennung, 1995*.
2. Adams, A., Baek, J., & Davis, M. A. Fast high-dimensional filtering using the permutohedral lattice. *In Computer Graphics Forum 29(2), 2010*.
3. Kopf, J. et al. Joint bilateral upsampling. *ACM Transactions on Graphics (TOG), 26(3), 2007*.
4. Krähenbühl, P., & Koltun, V. Efficient inference in fully Connected CRFs with Gaussian edge potentials. *In NIPS, 2011*.
5. Everingham, M. et al. The Pascal visual object classes (voc) challenge. *IJCV, 88(2), 2010.*
6. Bell, S. et al. Material recognition in the wild with the materials in context database. In *CVPR, 2015.*
7. Liang Chieh, C. et al. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *In ICLR*, 2015.

{varun.jampani, martin.kiefel, peter.gehler}@tuebingen.mpg.de

Perceiving Systems – ps.is.tue.mpg.de