Superpixel Convolutional Networks using Bilateral Inceptions

Raghudeep Gadde*[1], Varun Jampani*[1], Martin Kiefel[1,2], Daniel Kappler[1] & Peter V. Gehler[1,2]

[1]MPI for Intelligent Systems, Tübingen; [2]Bernstein Center for Computational Neuroscience, Tübingen

Eberhard Karls Universität Tübingen

We propose 'Bilateral Inception' module that propagates structured information in CNNs for segmentation.

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

Image Conditioned Filtering Inside CNNs

This work makes two contributions for image labeling CNNs:

1. Easy to adapt image conditioned filtering within CNN architectures.
2. Recovering arbitrary image resolutions of CNN outputs.

The proposed *Bilateral Inception* module implements the following prior information for segmentation.

- *Pixels that are spatially and photometrically similar are more likely to have the same label.*

In contrast to CNN/(Dense)CRF combinations, information is propagated directly *within* the CNN using image adaptive filters.

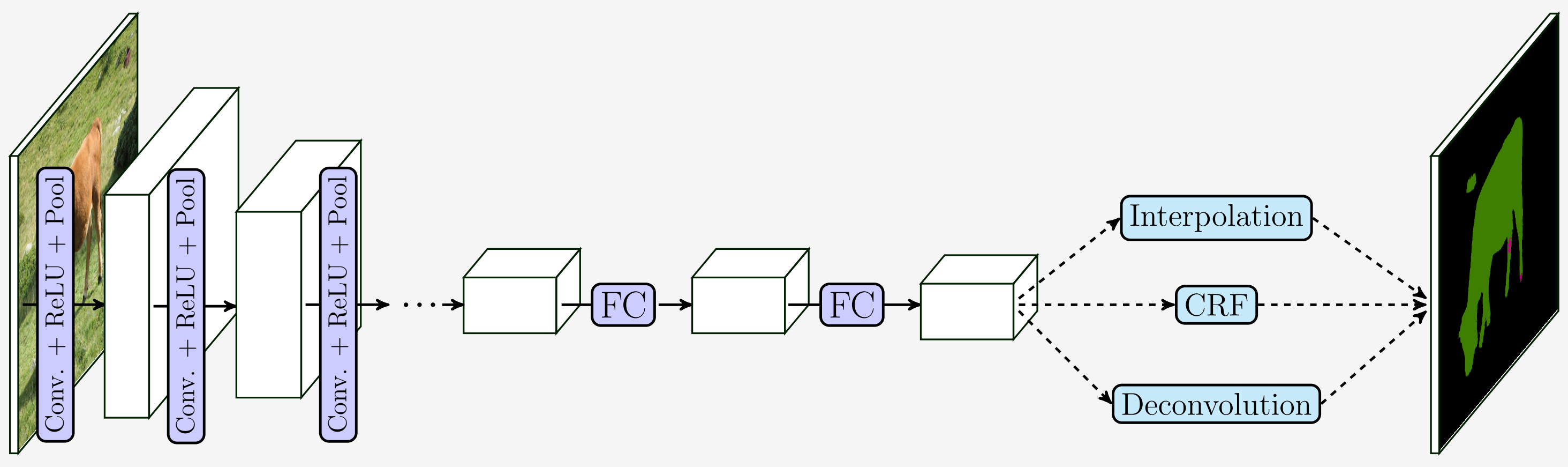


Fig.1: Different refining/upsampling strategies for segmentation CNNs

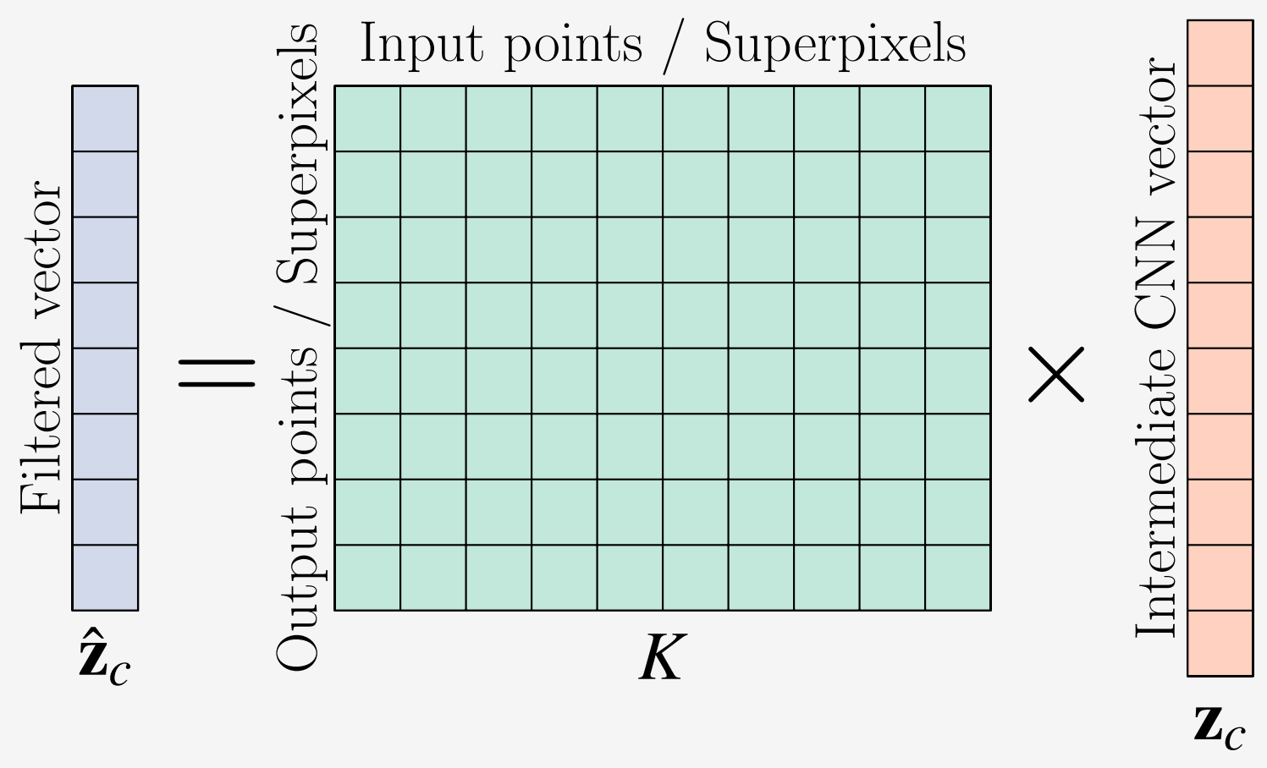
Bilateral Inception Module

Bilateral Filtering:

- Edge preserving filter [2] that works in high-dimensional feature spaces.
- Given input points with features F_{in} and output points with features F_{out}, Gaussian bilateral filtering an intermediate CNN representation $\mathbf{z}$ amounts to a matrix-vector multiplication, for each feature channel, c :

$$\hat{\mathbf{z}}_c = K(\theta, \Lambda, F_{in}, F_{out})\mathbf{z}_c$$

$$K_{i,j} = \frac{\exp(-\theta\|\Lambda \mathbf{f}_i - \Lambda \mathbf{f}_j\|^2)}{\sum_{j'} \exp(-\theta\|\Lambda \mathbf{f}_i - \Lambda \mathbf{f}_{j'}\|^2)}.$$



Λ: *Feature transformation matrix*; θ: *Filter scale*.

The *Bilateral Inception* module (BI) is a weighted combination of bilateral filters with different scales $\theta^1, \ldots, \theta^H$ (see Fig.2):

$$\bar{\mathbf{z}}_c = \sum_{h=1}^{H} \mathbf{w}_c^h \hat{\mathbf{z}}_c^h$$

Bilateral filtering is *modularly* implemented for the reuse of intermediate computations (see Fig.3).

Input/output points need *not* lie on a grid.

We use *superpixels* for computational reasons. Also results in *full-resolution output*.

All the free parameters for the BI module $\mathbf{w}, \{\theta^h\}$ and Λ are learned via backpropagation.

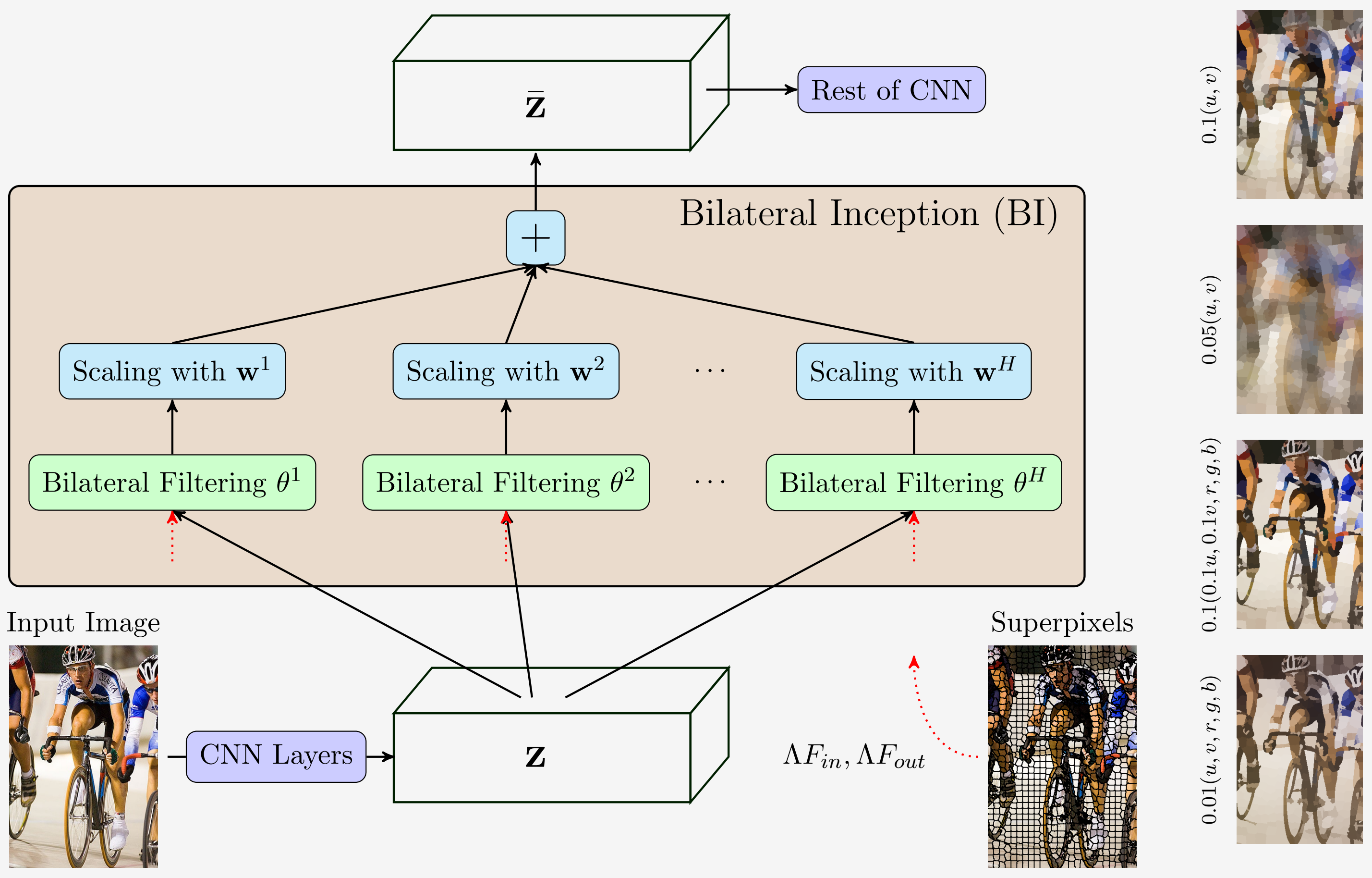

$\bar{\mathbf{z}}$
Rest of CNN
Bilateral Inception (BI)
+
Scaling with $\mathbf{w}^1$
Scaling with $\mathbf{w}^2$
···
Scaling with $\mathbf{w}^H$
Bilateral Filtering θ^1
Bilateral Filtering θ^2
···
Bilateral Filtering θ^H
Input Image
CNN Layers
$\mathbf{z}$
$\Lambda F_{in}, \Lambda F_{out}$
Superpixels
$0.1(u,v)$
$0.05(u,v)$
$0.1(0.1u, 0.1v, r, g, b)$
$0.01(u, v, r, g, b)$

Fig.2: Illustration of a bilateral inception (BI) module

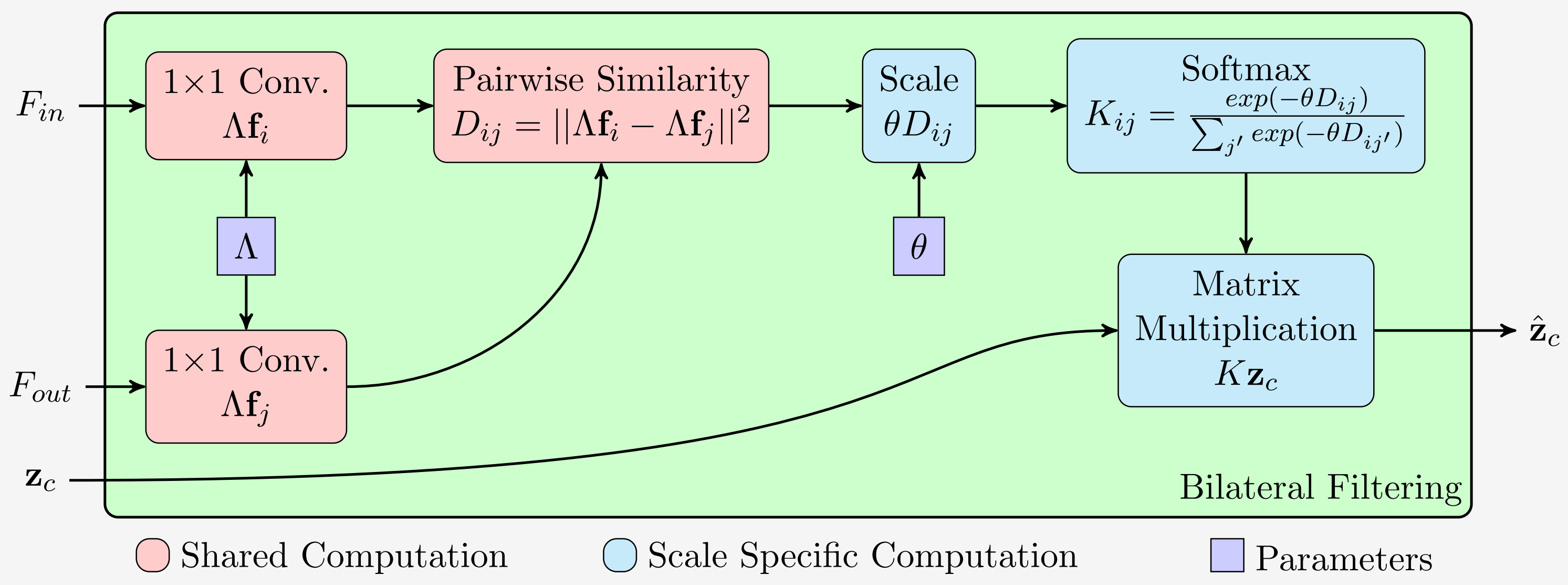

F_{in}
1×1 Conv. $\Lambda \mathbf{f}_i$
Pairwise Similarity $D_{ij} = ||\Lambda \mathbf{f}_i - \Lambda \mathbf{f}_j||^2$
Scale θD_{ij}
Softmax $K_{ij} = \frac{exp(-\theta D_{ij})}{\sum_{j'} exp(-\theta D_{ij'})}$
Λ
θ
Matrix Multiplication $K\mathbf{z}_c$
$\hat{\mathbf{z}}_c$
F_{out}
1×1 Conv. $\Lambda \mathbf{f}_j$
$\mathbf{z}_c$
Bilateral Filtering
Shared Computation
Scale Specific Computation
Parameters

Fig.3: Computation flow of the Gaussian bilateral filtering

Experiments

We insert BI modules between 1x1 convolution (FC) layers in standard CNN architectures.

$BI_k(H)$ indicates BI module after FC_k layer with H number of bilateral filters.

Experiments with 3 different architectures and on 3 different datasets:

Observations:

- BI modules reliably improve CNN performance with little overhead of time.
- In addition to producing sharp boundaries (like in DenseCRF), BI modules also help in better predictions due to information propagation between CNN units.
- Fast and effective in comparison to state-of-the-art dense pixel prediction techniques.

Generalization to different superpixel layouts

- BI modules are *flexible* in terms of number of input/output points.
- We observe that the BI networks trained with particular superpixel layout *generalize to other superpixel layouts* obtained with agglomerative hierarchical clustering.

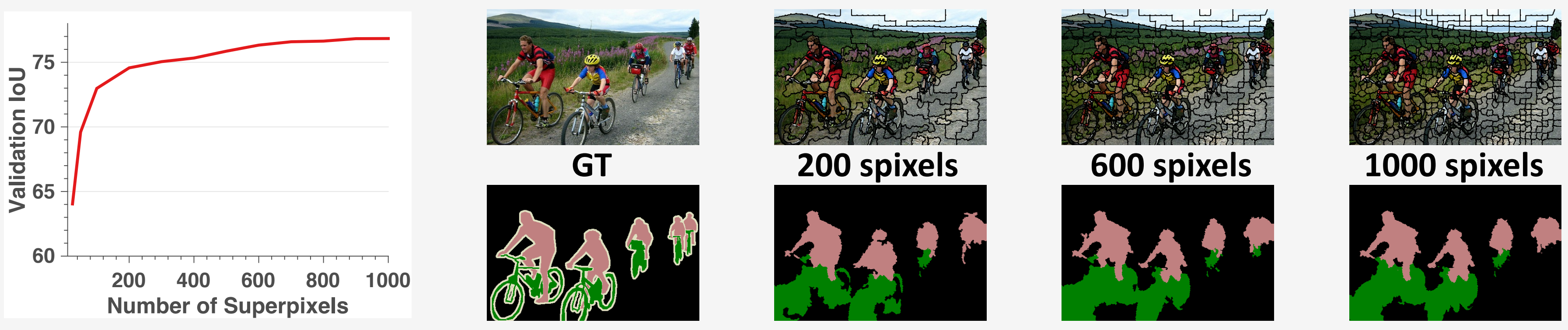


Fig.5: The effect of superpixel granularity on IoU.

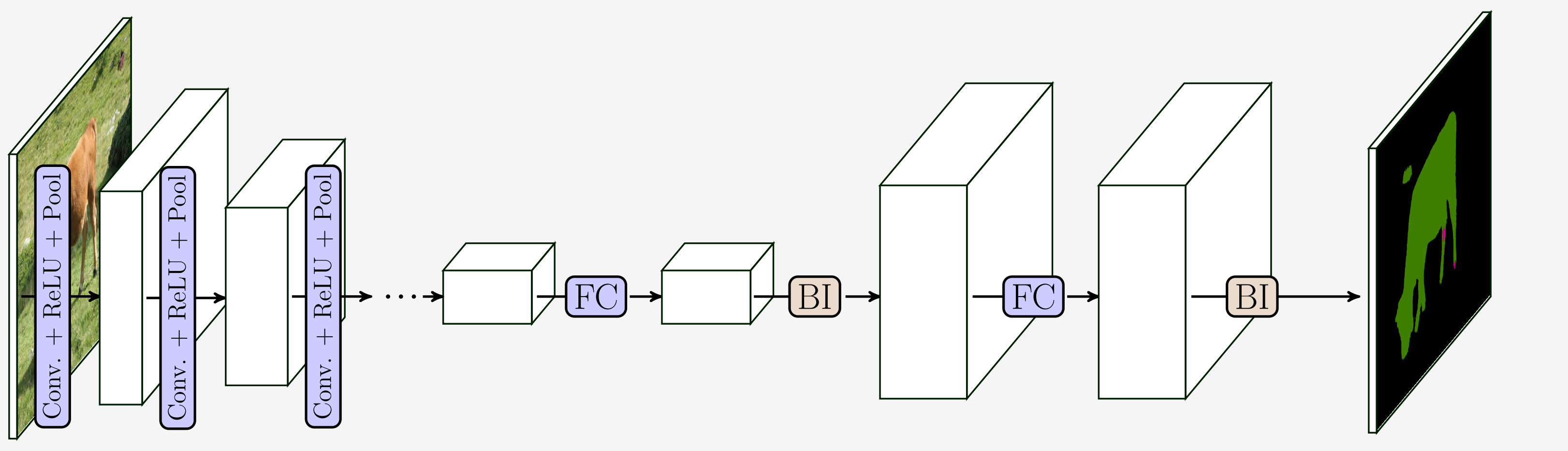


Fig.4: Segmentation CNN with bilateral inception (BI) modules

Model	Training	*IoU*	*Runtime*
DeepLab [5]		68.9	145ms
With BI modules			
$\mathrm{BI}_6(2)$	only BI	70.8	+20
$\mathrm{BI}_6(2)$	BI+FC	71.5	+20
$\mathrm{BI}_6(6)$	BI+FC	72.9	+45
$\mathrm{BI}_7(6)$	BI+FC	73.1	+50
$\mathrm{BI}_8(10)$	BI+FC	72.0	+30
$\mathrm{BI}_6(2)$-$\mathrm{BI}_7(6)$	BI+FC	73.6	+35
$\mathrm{BI}_7(6)$-$\mathrm{BI}_8(10)$	BI+FC	73.4	+55
$\mathrm{BI}_6(2)$-$\mathrm{BI}_7(6)$	FULL	**74.1**	+35
$\mathrm{BI}_6(2)$-$\mathrm{BI}_7(6)$-CRF	FULL	**75.1**	+865
DeepLab-CRF [5]		72.7	+830
DeepLab-MSc-CRF [5]		**73.6**	+880
DeepLab-EdgeNet [6]		71.7	+30
DeepLab-EdgeNet-CRF [6]		**73.6**	+860

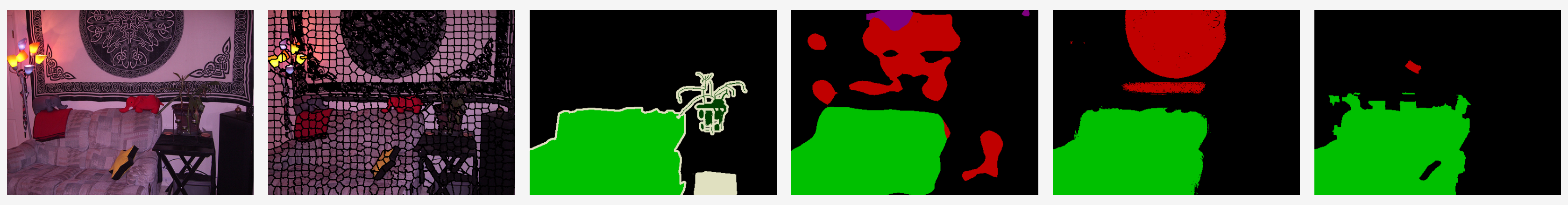
Tab.1: Results with DeepLab models on Pascal VOC12

Model	*IoU*	*Runtime*
DeconvNet(CNN+Deconv.) [7]	72.0	190ms
With BI modules		
$\mathrm{BI}_3(2)$-$\mathrm{BI}_4(2)$-$\mathrm{BI}_6(2)$-$\mathrm{BI}_7(2)$	**74.9**	245
CRFasRNN (DeconvNet-CRF) [7]	74.7	2700

Tab.2: Results with CRFasRNN models on Pascal VOC12

Model	Class / Total accuracy	Runtime
Alexnet CNN [4]	55.3 / 58.9	300ms
$\mathrm{BI}_7(2)$-$\mathrm{BI}_8(6)$	67.7 / 71.3	410
$\mathrm{BI}_7(6)$-$\mathrm{BI}_8(6)$	**69.4 / 72.8**	470
AlexNet-CRF [4]	65.5 / 71.0	3400

Tab.3: Results with Alexnet models on MINC material segmentation dataset



Input Image Superpixels GT DeepLab CNN + DenseCRF With BI

Fig.6: Example visual results of semantic segmentation on Pascal VOC12 dataset image.

Conclusion

Bilateral Inception models aim to directly include the model structure of CRF factors into the forward architecture of CNNs. They are fast, easy to implement and can be inserted into existing CNN models.

References:

1. Krähenbühl, P., & Koltun, V. Efficient inference in fully Connected CRFs with Gaussian edge potentials. *In NIPS, 2011*.
2. Aurich, V., & Weule, J. Non-linear Gaussian filters performing edge preserving diffusion. *In Mustererkennung, 1995*.
3. Everingham, M. et al. The Pascal visual object classes (voc) challenge. *IJCV, 88(2), 2010.*
4. Bell, S. et al. Material recognition in the wild with the materials in context database. In *CVPR, 2015.*
5. Liang Chieh, C. et al. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *In ICLR*, 2015.
6. Liang Chieh, C. et al. Semantic Image Segmentation with Task-Specific Edge Detection Using CNNs and a Discriminatively Trained Domain Transform, *In CVPR*, 2016.
7. Zheng, S. et al. Conditional Random Fields as Recurrent Neural Networks, *In ICCV*, 2015.
8. Cordts, M. et al. The Cityscapes Dataset for Semantic Urban Scene Understanding, *In CVPR,* 2016.

*Joint first authors {raghudeep.gadde, varun.jampani, martin.kiefel, daniel.kappler, peter.gehler}@tuebingen.mpg.de