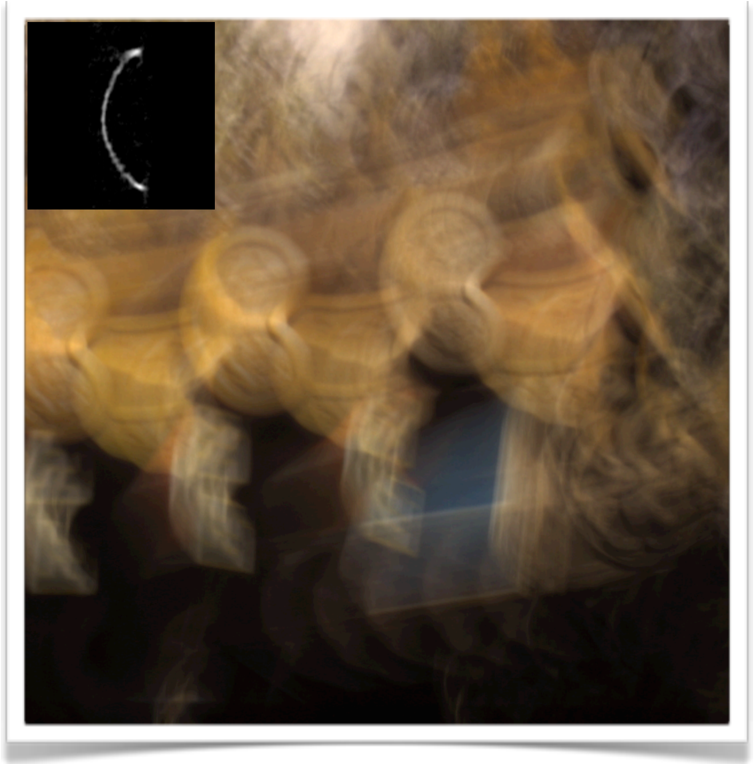
Discriminative Non-blind Deblurring



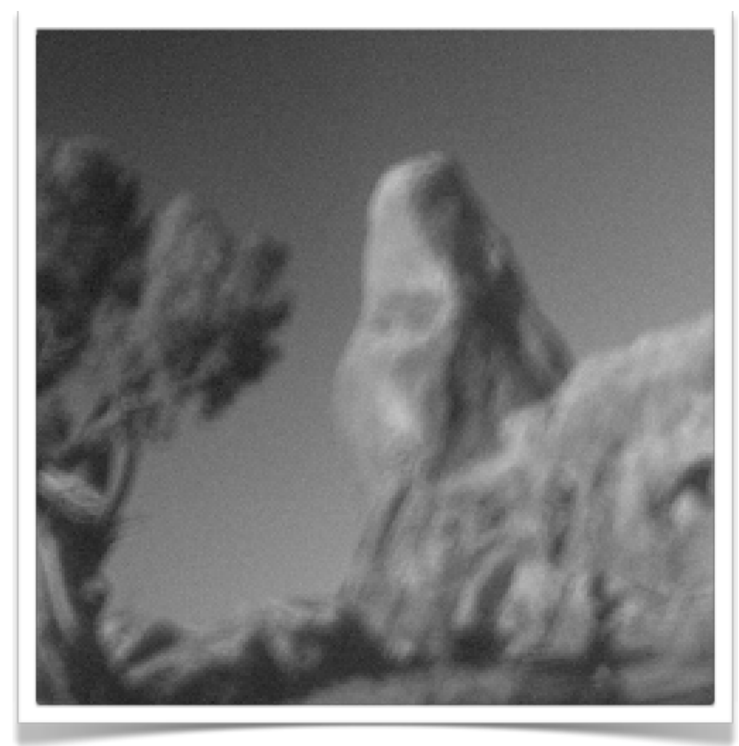
U. Schmidt[1], C. Rother[2], S. Nowozin[2], J. Jancsary[2], S. Roth[1]

1 TECHNISCHE UNIVERSITÄT DARMSTADT

2 Microsoft® Research

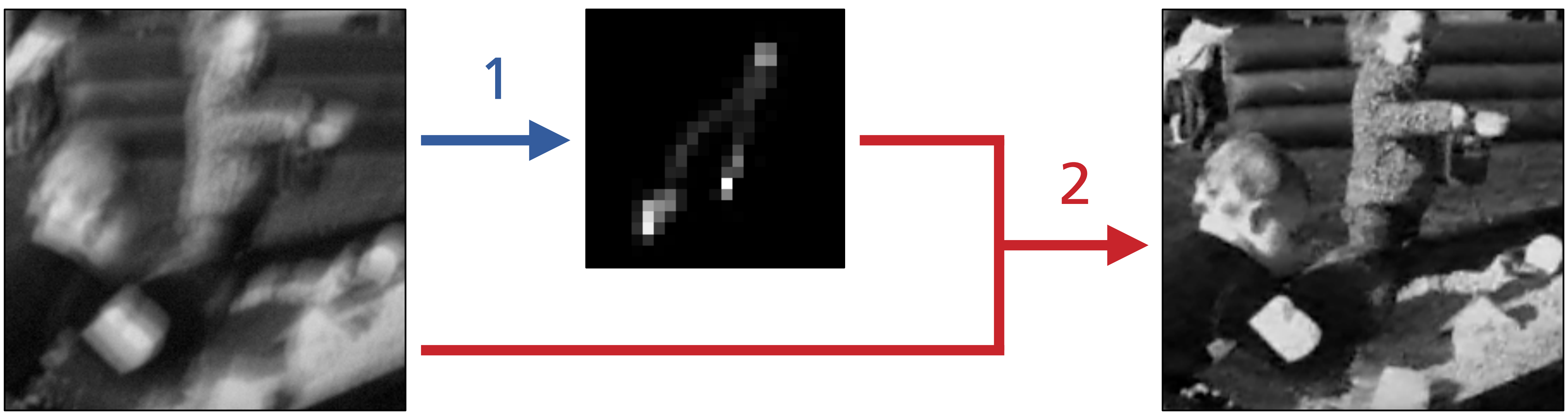
Image Blur

- Sources of image blur
  - camera motion
  - objects out-of-focus
- Why remove?
  - restoring digital photographs
  - to cope with adverse imaging conditions
- Why difficult?
  - loss of information, especially high frequencies
  - mathematically ill-posed



Single Image Deblurring

- 1st step: blur estimation
  [e.g. Fergus et al. '06; Whyte et al. '10; Levin et al. '11]
  - blur assumption: uniform vs. non-uniform
- 2nd step: **non-blind deblurring** using blur estimate
  - popular approach: using **image priors**
    [e.g. Levin et al. '07; Krishnan and Fergus '09; Schmidt et al. '11]



Previous work: Prior-based Deblurring

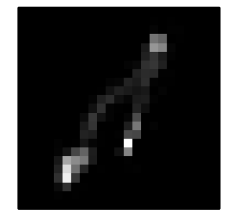
Modeling assumptions:

Blurred $\mathbf{y}$

=

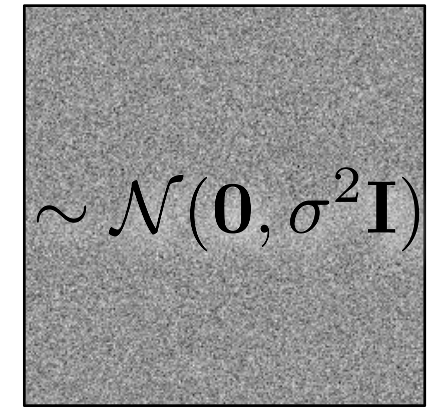
Clean image $\mathbf{x}$

$\otimes$



Kernel $\mathbf{k}$

+



Gaussian noise

Typical solution approach:

Deblurred $\hat{\mathbf{x}}$

$\longleftarrow$ *arg max* POSTERIOR $\propto$ LIKELIHOOD $\cdot$ PRIOR$^{\lambda}$

- Likelihood is **fixed**
- Prior is **manually chosen**
- Regularization weight λ **tuned on validation set**

Limited flexibility → learning-based approach attractive

Discriminative Non-blind Deblurring

- **Flexible learning-based approach with competitive runtime**
  - generative approach of [Schmidt et al. '11] computationally expensive

  → discriminative *conditional random field* (CRF) model

- **Three main challenges to overcome:**

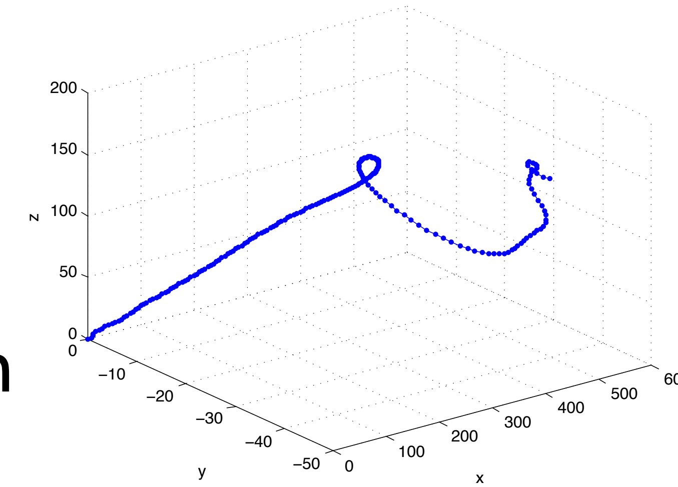
1. Lack of training data, in particular realistic blur kernels
2. Adapt model based on blurred image content
3. Train model to work with arbitrary images *and* blurs

   → blur at test time not known during training

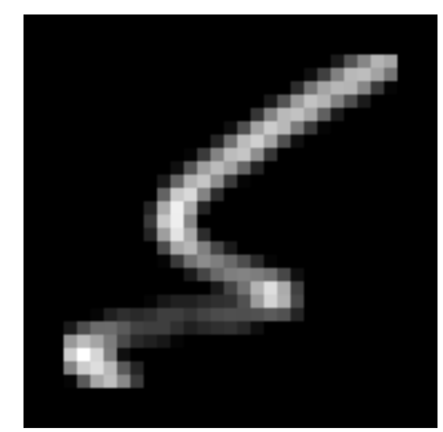
➡ **Discriminative deblurring with state-of-the-art performance**

Overcoming Challenges (1)

- Realistic blur kernels are scarce
  - recording them is difficult
  - existing ones used for testing, shouldn't be trained on



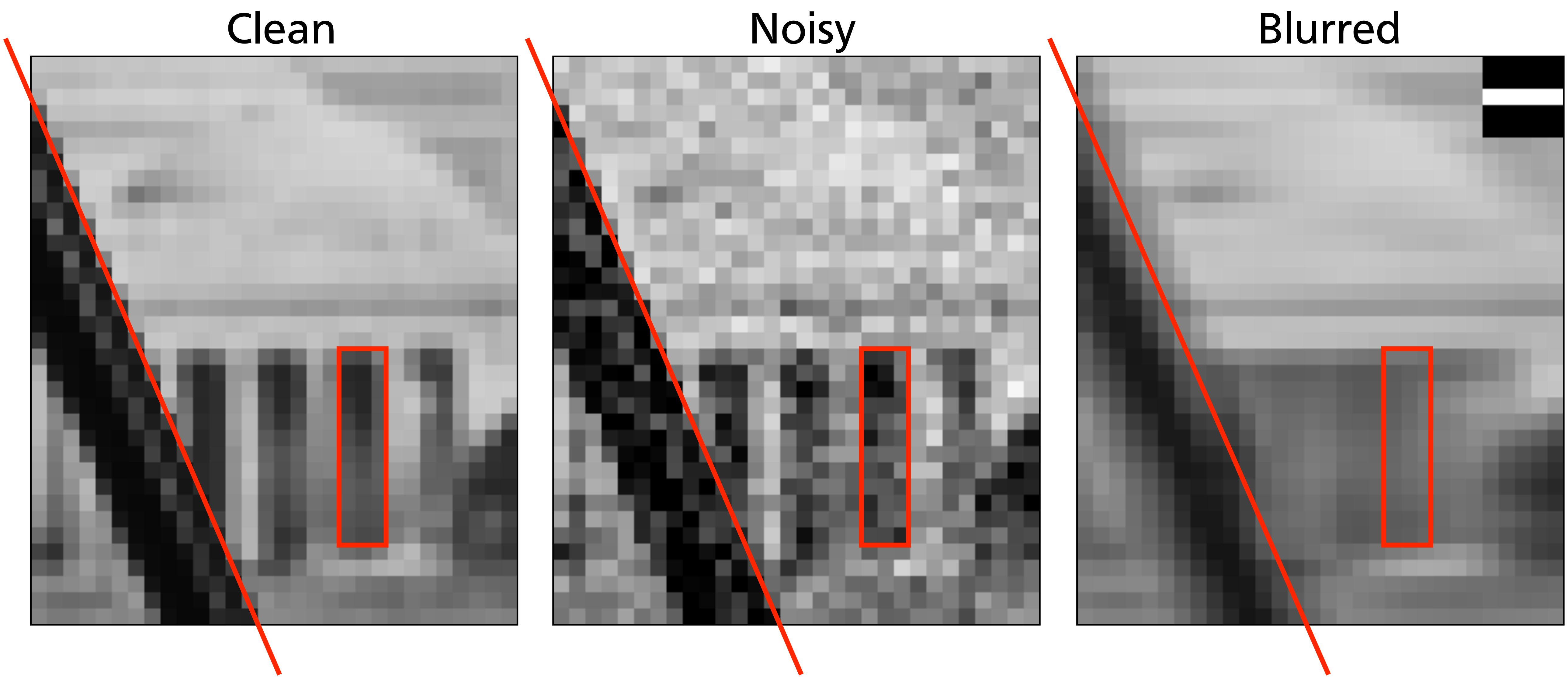
- **Generate artificial blur kernels**
  - from random 3D trajectories with simple motion model



- Synthesize training data from clean images and blur kernels

Overcoming Challenges (2)

- How to adapt model based on observed image?
  - difficult due to blurred image content
  - easier for denoising → Gaussian CRF [Tappen et al. '07, Jancsary et al. '12]



Overcoming Challenges (2)

- How to adapt model based on observed image?
  - difficult due to blurred image content

- **Half-quadratic deblurring** commonly used with image priors
  [e.g. Levin et al. '07; Krishnan and Fergus '09]
  - introduce latent variables to make inference easier

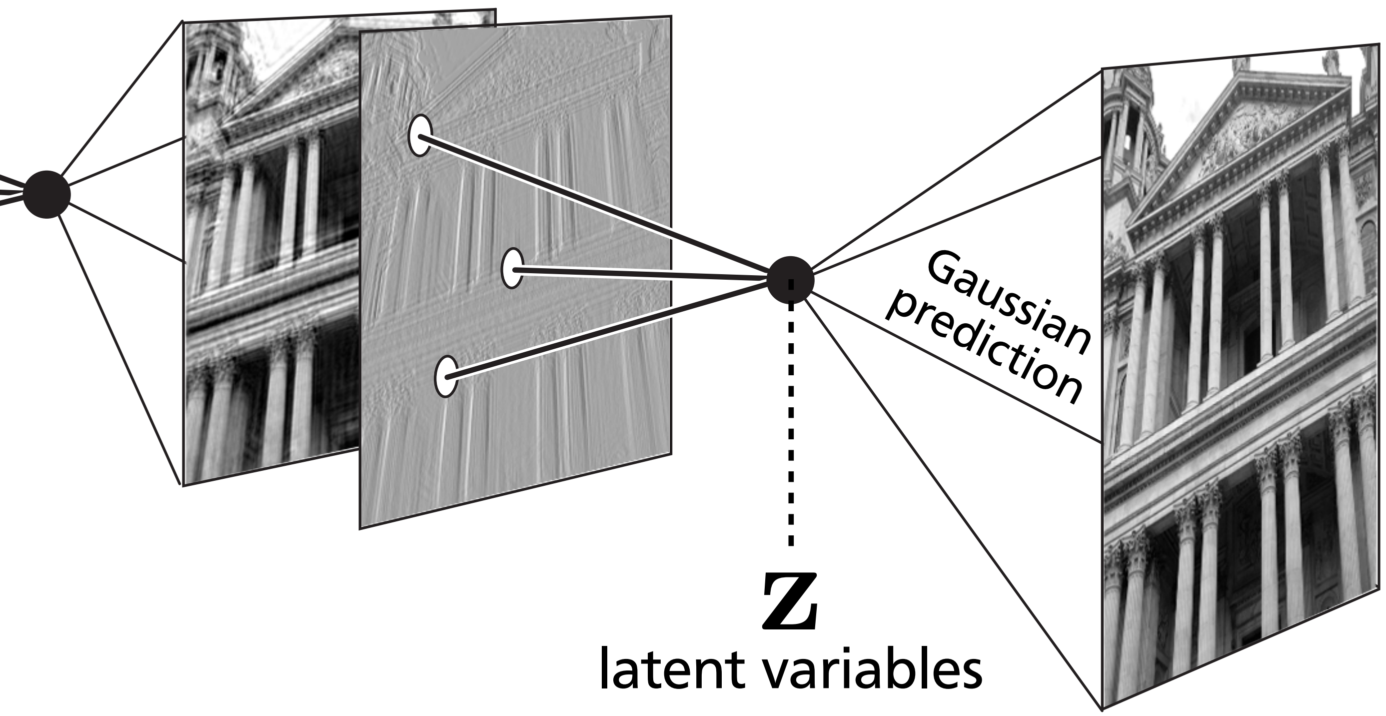
Update deblurred image ⇄ Iterate ⇄ Update latent variables

Half-quadratic MAP estimation

Half-Q. vs. Discriminative Cascade

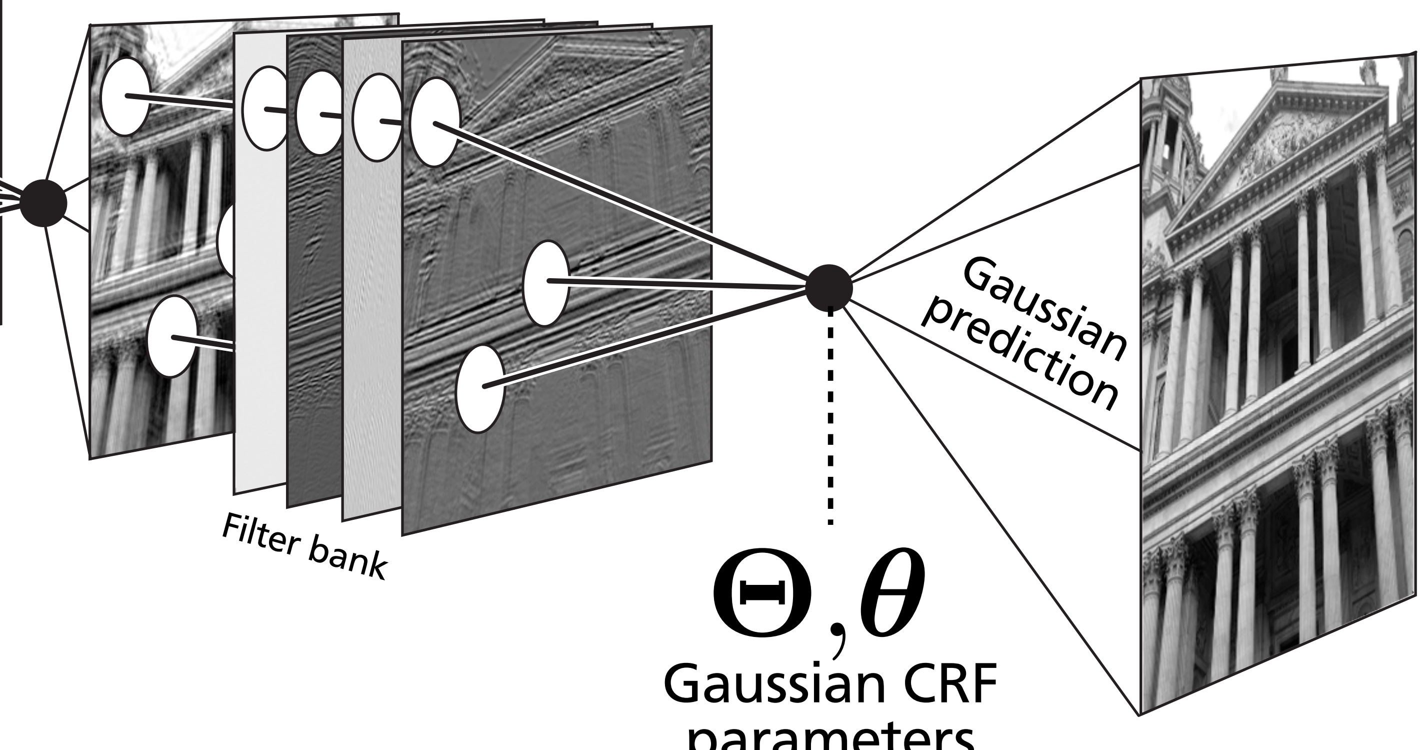
Half-quadratic deblurring

- latent variables define inhomogeneous Gaussian MRF
- **restricted update** iterations (based on image prior)



Discriminative cascade

- latent variables replaced by **trained Gaussian CRF**
- can use arbitrary features
  → more flexible
  → better in fewer iterations



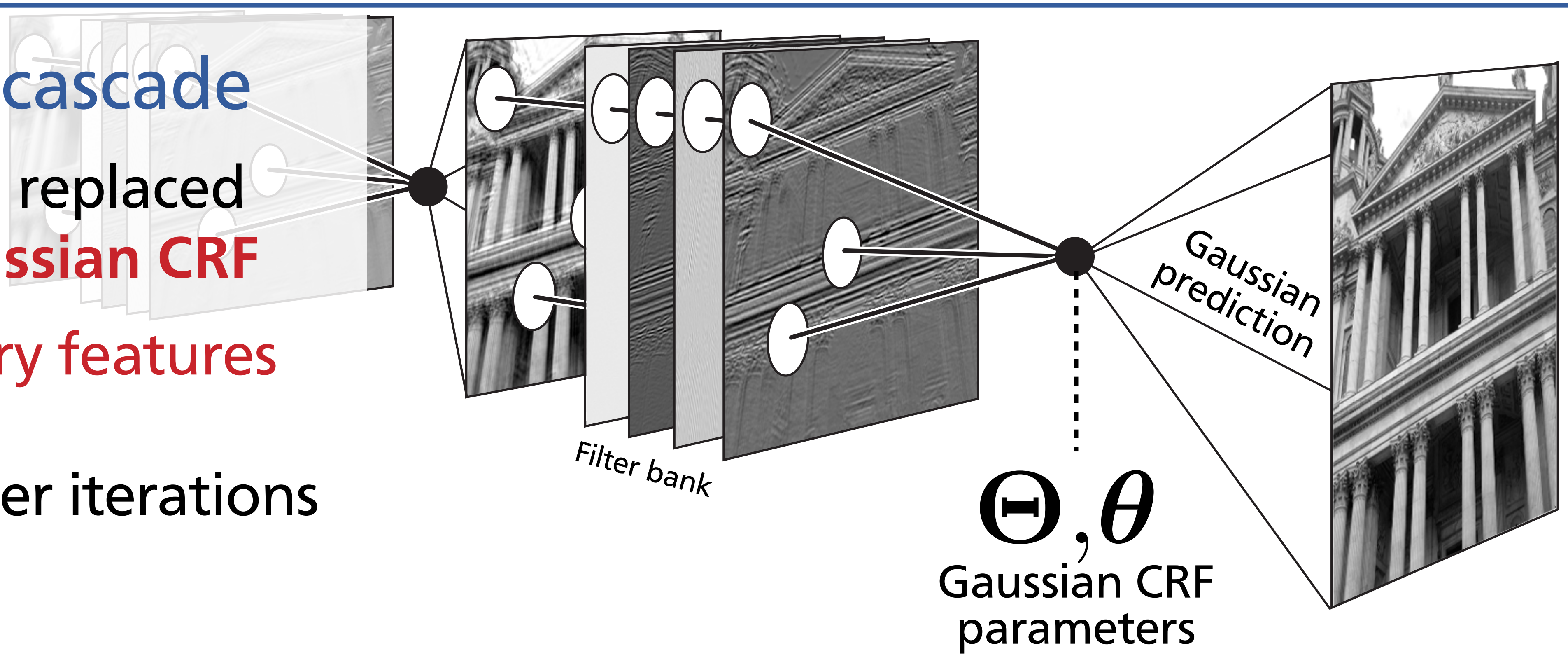
Half-Q. vs. Discriminative Cascade

Microsoft Research
TECHNISCHE UNIVERSITÄT DARMSTADT

- Discriminative cascade generalizes half-quadratic deblurring
- Previous Gaussian CRFs → one stage of proposed cascade
  - single stage sufficient for simpler tasks (e.g. image denoising)
  - would likely benefit from iterative refinement

Discriminative cascade

- latent variables replaced by **trained Gaussian CRF**
- can use arbitrary features
  → more flexible
  → better in fewer iterations



Overcoming Challenges (3)

- Learn model that works with arbitrary images *and* blurs
- **Idea:** split model parameters into learnable and blur-dependent ones
  - akin to combining prior and likelihood in a generative approach

$$p(\mathbf{x}|\mathbf{y},\mathbf{K}) \propto \underbrace{\mathcal{N}\left(\mathbf{x}; (\alpha\mathbf{K}^{\mathrm{T}}\mathbf{K})^{-1}\alpha\mathbf{K}^{\mathrm{T}}\mathbf{y}, (\alpha\mathbf{K}^{\mathrm{T}}\mathbf{K})^{-1}\right)}_{\text{Likelihood}} \cdot \underbrace{\mathcal{N}\left(\mathbf{x}; \boldsymbol{\Theta}^{-1}\boldsymbol{\theta}, \boldsymbol{\Theta}^{-1}\right)}_{\text{Prior}}$$

$$\propto \underbrace{\mathcal{N}\left(\mathbf{x}; (\boldsymbol{\Theta} + \alpha\mathbf{K}^{\mathrm{T}}\mathbf{K})^{-1}(\boldsymbol{\theta} + \alpha\mathbf{K}^{\mathrm{T}}\mathbf{y}), (\boldsymbol{\Theta} + \alpha\mathbf{K}^{\mathrm{T}}\mathbf{K})^{-1}\right)}_{\text{Posterior}}$$

Overcoming Challenges (3)

- Learn model that works with arbitrary images *and* blurs
- **Idea:** split model parameters into learnable and blur-dependent ones
  - akin to combining prior and likelihood in a generative approach

$$p(\mathbf{x}|\mathbf{y},\mathbf{K}) \propto \underbrace{\mathcal{N}\left(\mathbf{x}; (\alpha\mathbf{K}^{\mathrm{T}}\mathbf{K})^{-1}\alpha\mathbf{K}^{\mathrm{T}}\mathbf{y}, (\alpha\mathbf{K}^{\mathrm{T}}\mathbf{K})^{-1}\right)}_{\text{Likelihood}} \cdot \underbrace{\mathcal{N}\left(\mathbf{x}; \boldsymbol{\Theta}^{-1}\boldsymbol{\theta}, \boldsymbol{\Theta}^{-1}\right)}_{\text{Prior}}$$

$$\propto \mathcal{N}\left(\mathbf{x}; (\boldsymbol{\Theta} + \alpha\mathbf{K}^{\mathrm{T}}\mathbf{K})^{-1}(\boldsymbol{\theta} + \alpha\mathbf{K}^{\mathrm{T}}\mathbf{y}), (\boldsymbol{\Theta} + \alpha\mathbf{K}^{\mathrm{T}}\mathbf{K})^{-1}\right)$$

$\boldsymbol{\Theta}(\mathbf{y})$ $\boldsymbol{\theta}(\mathbf{y})$ $\boldsymbol{\Theta}(\mathbf{y})$

Adapted via learned regression function

→ Gaussian CRF for deblurring

Regression Tree Fields (RTFs)

- Flexible Gaussian CRF [Jancsary et al., CVPR'12]
  - non-linear regression of $\boldsymbol{\Theta}(\mathbf{y})$ and $\boldsymbol{\theta}(\mathbf{y})$ (via regression trees)

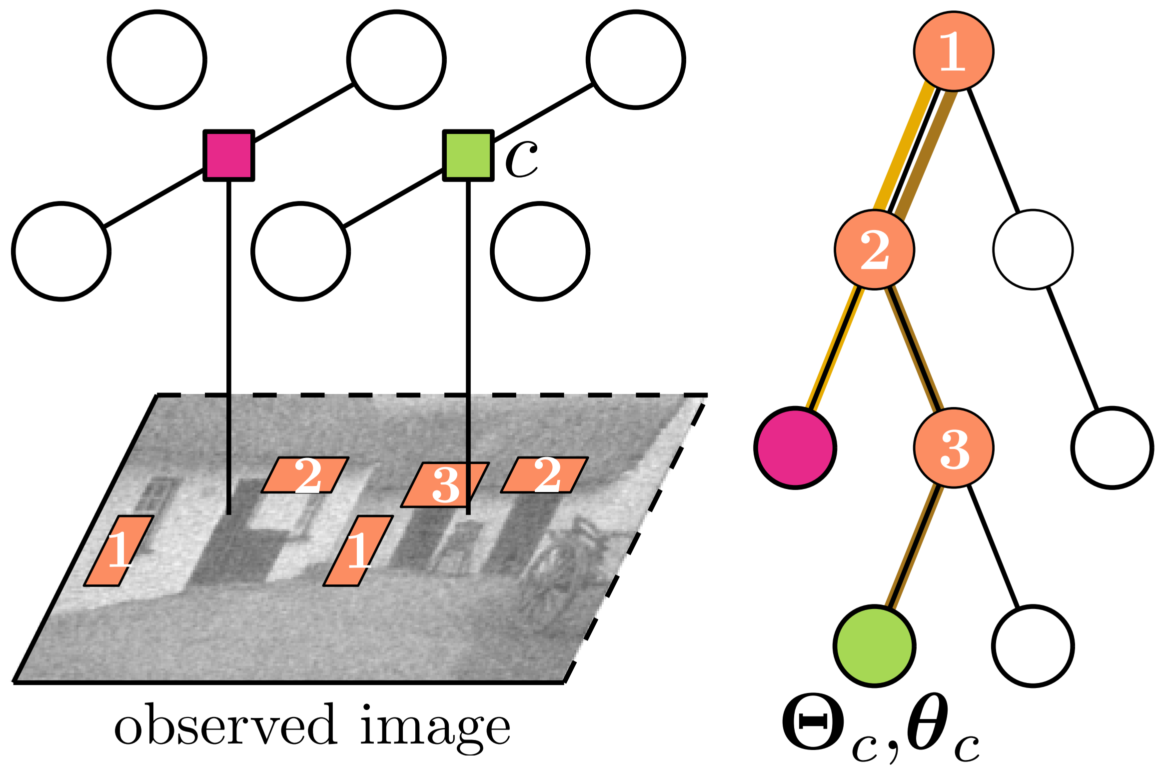
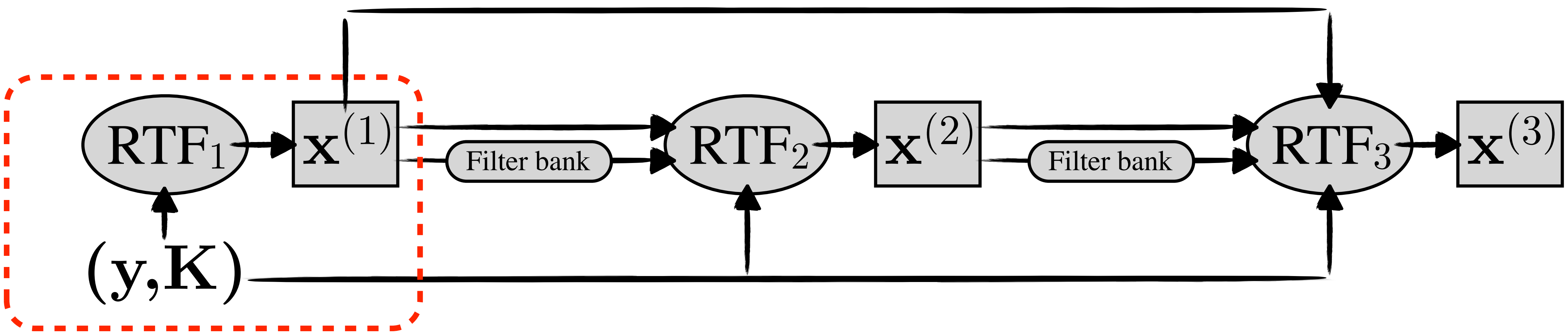


Figure adapted from [Jancsary et al., ECCV'12]

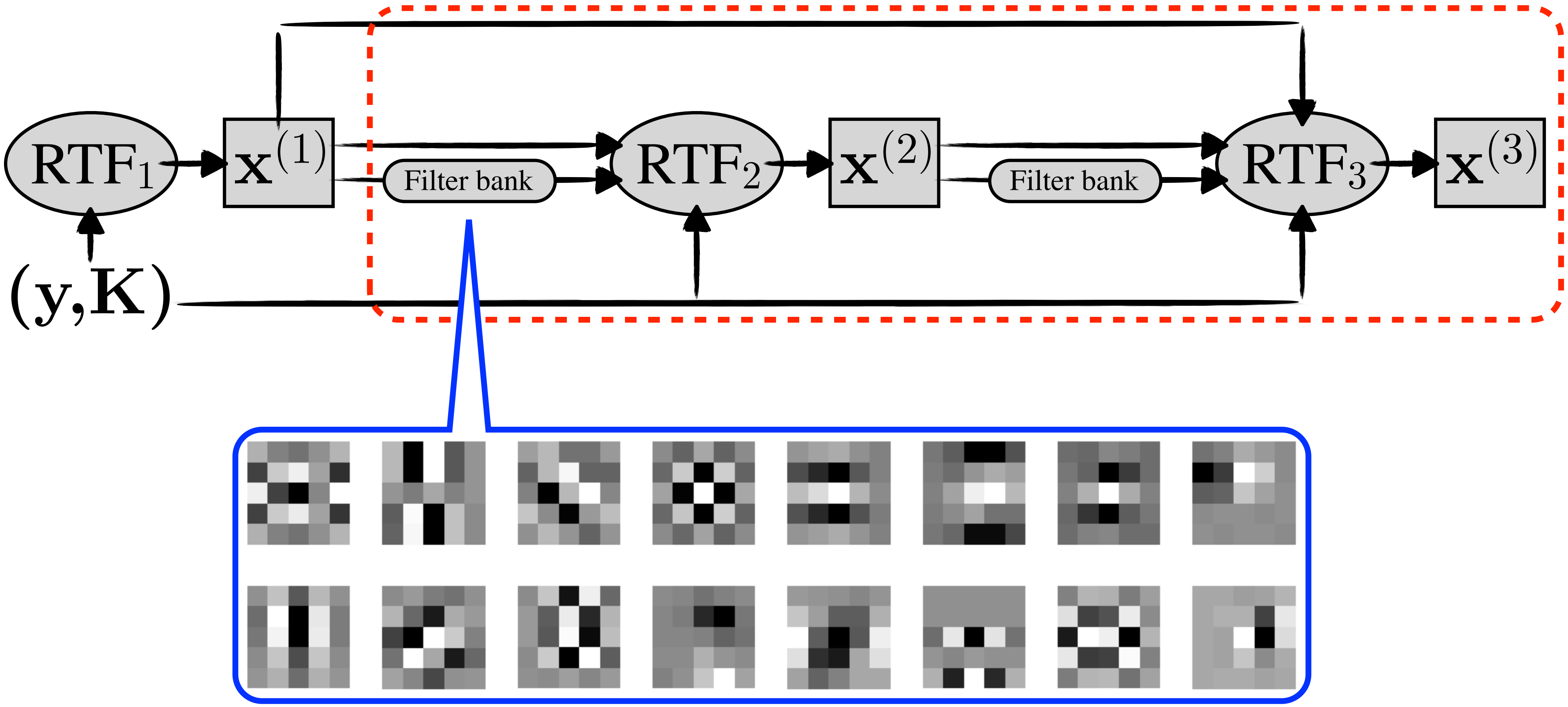
- Loss-based training (optimizing PSNR) [Jancsary et al., ECCV'12]
- Extend previous RTFs
  - 1) incorporating blur parameters, 2) using a model cascade

RTF Prediction Cascade



- 1^{st} stage RTF_1
  - blurred image is only feature, no trees used
  - crude estimate of restored image, already good when noise small

RTF Prediction Cascade



- Stages RTF_n
  - results of previous stages as input + filter responses
  - generatively-trained filter bank of [Gao and Roth '12]

Example of Model Stages

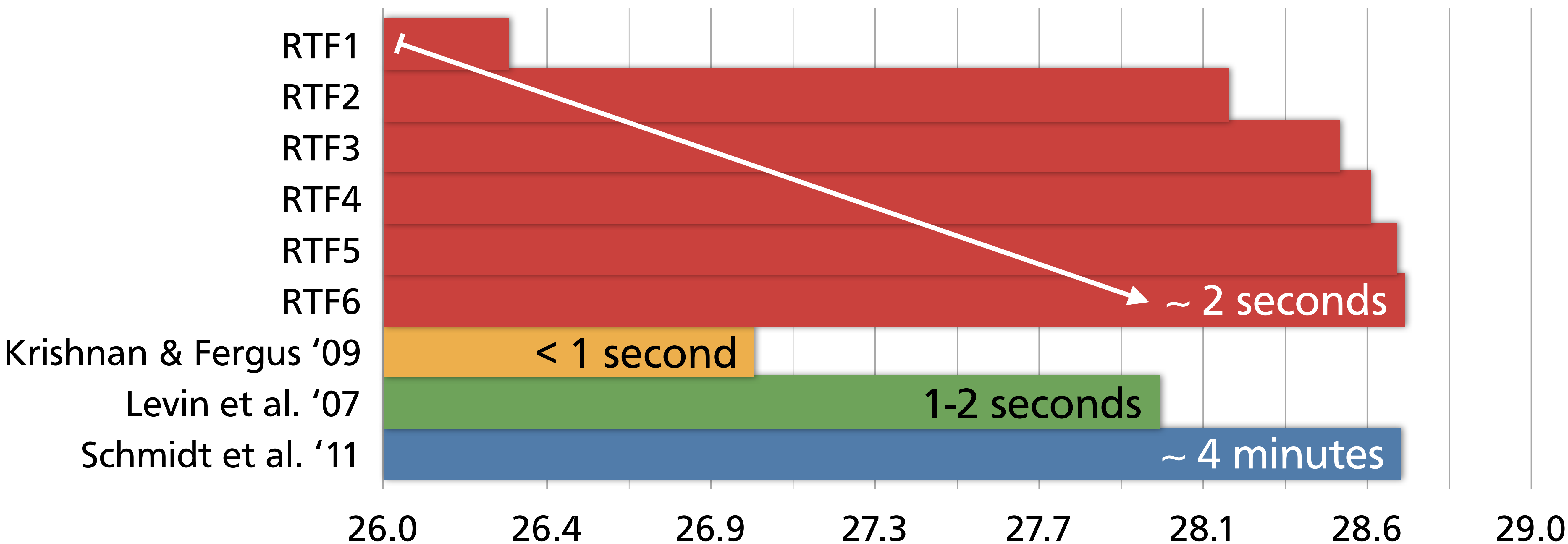
Ground truth

Blurred

RTF_1, 25.39dB → RTF_2, 27.71dB ··· → RTF_6, 28.20dB

Experiments (1)

- Benchmark with synthetically blurred images
- Trained 6-stage RTF cascade (with Gaussian noise σ = 2.55)

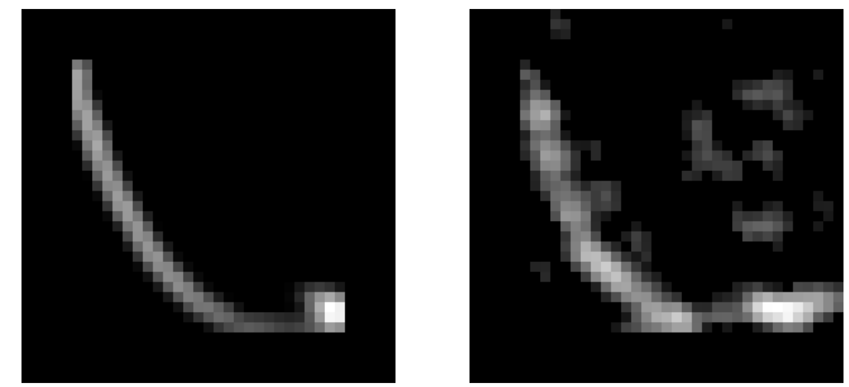


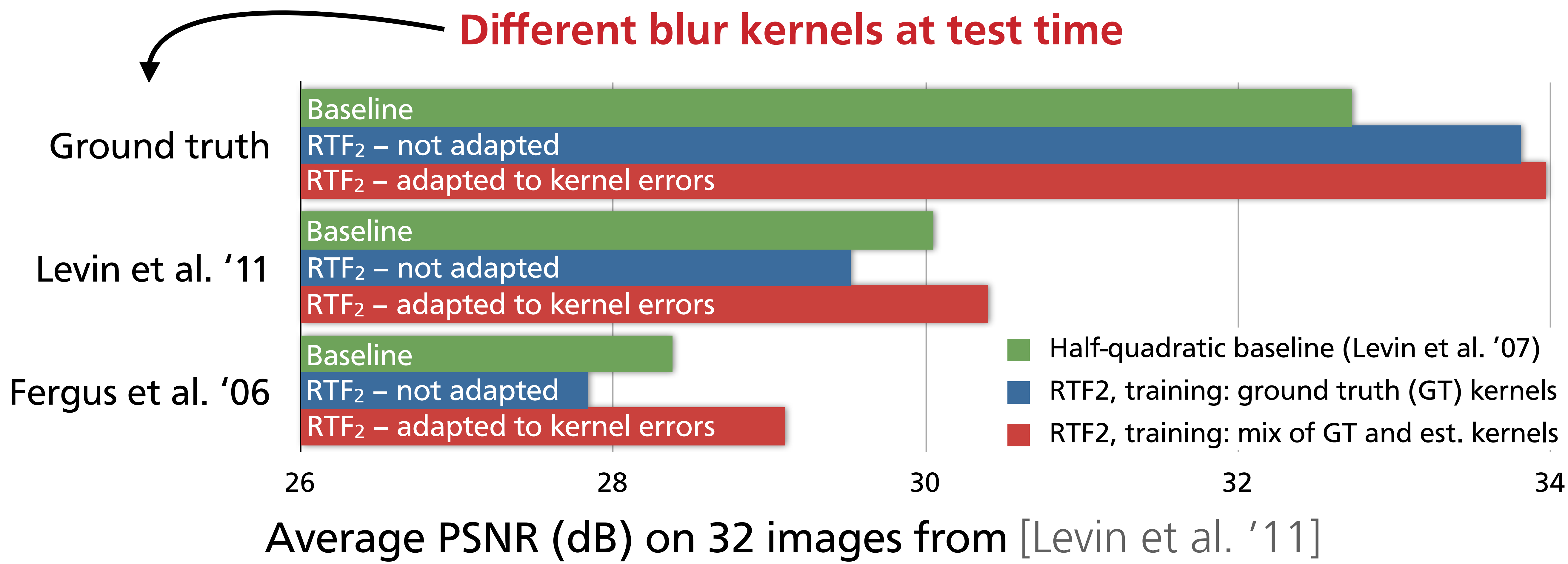
Average PSNR (dB) on 64 images from [Schmidt et al. '11]

Experiments (2)

- Adaptation to kernel estimation errors in blind deblurring
- Trained 2-stage RTF cascade
  (with Gaussian noise $\sigma = 0.5$)

Ground truth | Estimated



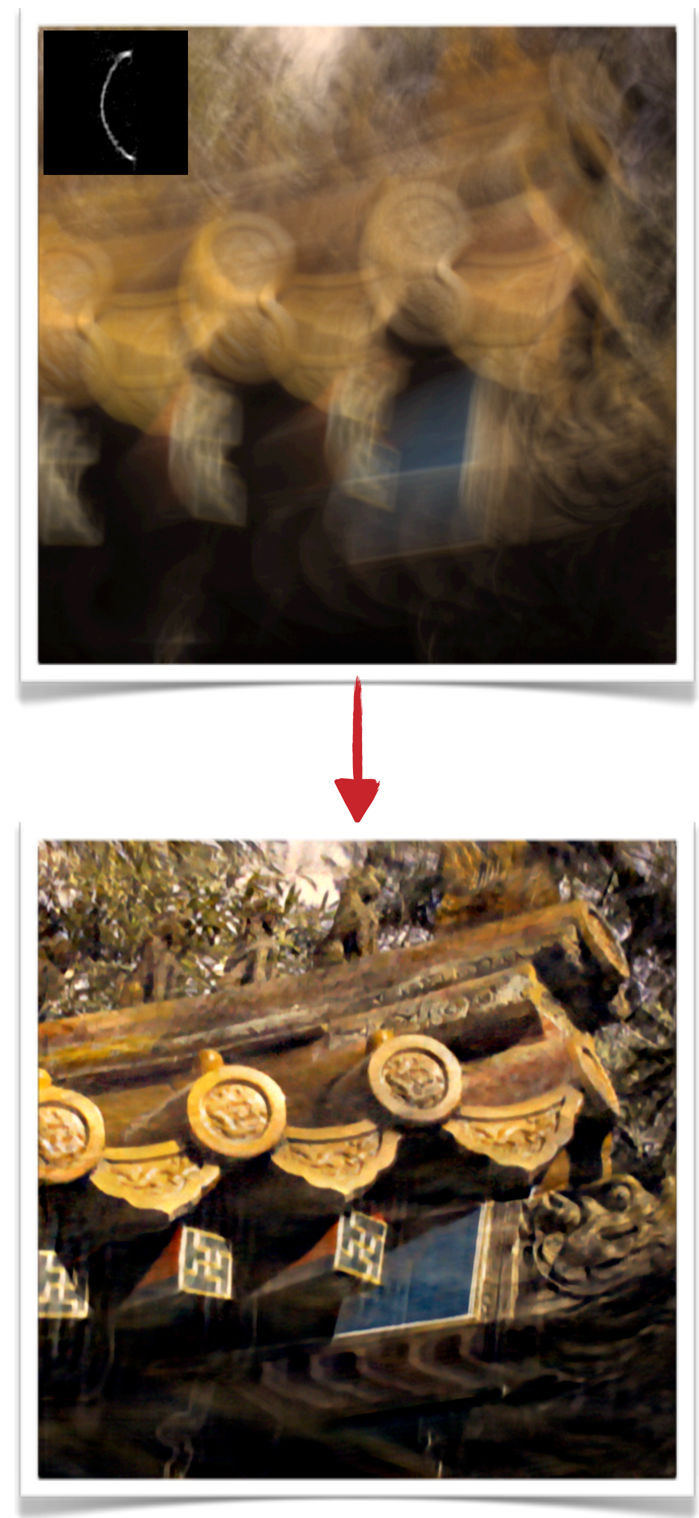


Experiments (3)

- Realistic higher-resolution images
- Can improve existing deblurring pipelines

	Image 1	Image 2	Image 3	Image 4
Kernel 01	+0.44	+0.54	+1.05	+0.76
Kernel 02	+0.44	+0.27	+0.38	+0.46
Kernel 03	+0.02	+0.03	+0.39	-0.26
Kernel 04	+0.31	+0.30	+0.61	+0.27
Kernel 05	+0.61	+0.44	+0.64	+0.05
Kernel 06	+0.40	+0.41	+1.03	+0.48
Kernel 07	+0.24	+0.55	+0.45	+0.31
Kernel 08	+0.76	+0.56	+2.17	+1.73
Kernel 09	+0.35	-0.09	+0.02	+0.23
Kernel 10	+0.19	-0.55	+0.25	+0.29
Kernel 11	-0.19	-0.43	+0.46	+0.09
Kernel 12	+0.76	+0.04	+0.66	+0.64

Improvement over results from [Xu and Jia '10]
(on avg. 0.41dB) in benchmark of [Köhler et al. '12]



Summary

- First discriminative non-blind deblurring approach
  - for arbitrary images and blurs
  - generalizes common half-quadratic deblurring
- Cascade model based on RTFs
  - loss-based training with synthesized data, including blurs
- State-of-the-art performance on three benchmarks
  - competitive runtime
- Proposed cascade not limited to image deblurring

Acknowledgements

- We thank **Pushmeet Kohli** for suggesting the topic of discriminative deblurring using a non-parametric model like the RTF.

- Funding from Microsoft Research PhD Scholarship Programme

Thank you for your attention.

Come to our poster