# ENGN 8501

# Week#2:  Low-level vision-2:

# Camera shake removal

2021 S2

# This week:

- Monday  Lecture:   Supervised paper reading
- Thursday tutorial-1:   Read an extra paper on the topic of l-lvl vision.
- Reading Report-1:  due on next Thursday evening.

# Enrollment update

- The current enrollment number is far beyond our prediction:
  - 45 → 65 (last week) → 91.
- This is an advanced level course, more demanding than 6528.
- Subjective assessments: reading report, project report, seminar.
  - *To speak a fact: if the class average mark is 66%, half of the students will get a mark below 66%.*

# Assessment Components: Your portfolio for ENGN8501

- **5x Paper Reading Reports:**
  - 2 pages each → totally 10 pages
  - 10% x 5 = 50% marks.
  - Must be in CVPR'21 Latex format. (otherwise, penalty applies)
  - → upload a single PDF file each report to Wattle. ( PDF is the only accepted file format.)

- **1x Group Research Project** (1~3 students per group):
  - → 1 page proposal : 5% marks
  - → up to 6 pages report (excluding references): 15% marks
  - → 10-minute video seminar recording and 10-slide PPT: 20% marks.

  - *Ideally, your 6-page project report, if novel and worthy enough, ought to be publishable at a local/national conference.*

# Homework assignment-1:
# Paper reading report #1

- Pick up any one paper from the first two weeks' home-reading materials ( see "Weekly-Study-Plan");
- Read the paper carefully and critically. If necessary, read also some of the references therein.
- Write a 2-page "reading report" using the provided latex template, and upload your report as a single PDF file to Wattle. (via TurnItIn).
- Your report may follow the general latex template.
- Your report must contain a "Critical Analysis" section (c.f. the latex template).
- . The marking process will be subject to tutors' (and lecturer's) best judgement. The class average mark might be somewhere at 5~6 out of 10.

# Outline for today's lecture

- 1. general tech-background introduction:  20 minutes
- 2. supervised paper reading session:   about 30 hours
- 3. breakout room: group discussion and report back to all: about 40 minutes.

# General Technical introduction

# Blind Image Deblur

# Generative model (forward process)

- original $f(x,y)$
- motion blur
- additive intensity noise

For an image with $n$ pixels, write this process as

$$\hat{\mathbf{g}} = \mathtt{A}\mathbf{f} + \mathbf{n}$$

where $\hat{\mathbf{g}}$ and $\mathbf{f}$ are $n$-vectors, and $\mathtt{A}$ is an $n \times n$ matrix.

# Inverse problem

- Estimate *f(x,y)* by optimizing a cost function:

$$\hat{\mathbf{f}} = \arg\min_{\mathbf{f}} \underbrace{(\overset{\text{observed image}}{\mathbf{g}} - \overset{\text{generated image}}{\mathtt{A}\mathbf{f}})^2}_{\text{Likelihood/ loss function}} + \lambda \underbrace{p(\mathbf{f})}_{\text{prior/ regularization}}$$

Example

$$p(f) = (\nabla \mathbf{f})^2$$

to suppress high frequency noise

# Blind deblurring Non-examinable

So far we have assumed that we know the generative model, e.g.

$$\mathbf{g} = A(\mathbf{h})\,\mathbf{f}$$

$$G = H\,F$$

$=$

$*$

i.e. that $h(x,y)$ is known, so that given the observed image $g(x,y)$, then the original image $f(x,y)$ can be estimated (restored)

Consider if only the observed image $g(x,y)$ is known. This is the problem of blind estimation.

# Blind deblurring continued

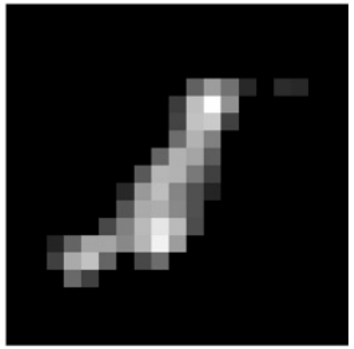
- Estimate *f(x,y)* and *h(x,y)* by optimizing a cost function:

$$\min_{\mathbf{f},\mathbf{h}} \underbrace{(\overset{\text{observed image}}{\mathbf{g}} - \overbrace{\mathtt{A}(\mathbf{h})\,\mathbf{f}}^{\text{generated image}})^2}_{\text{Likelihood/ loss function}} + \lambda \underbrace{p_f(\mathbf{f})}_{\text{image prior}} + \mu \underbrace{p_h(\mathbf{h})}_{\text{blur prior}}$$

# Example I: Blind deblurring

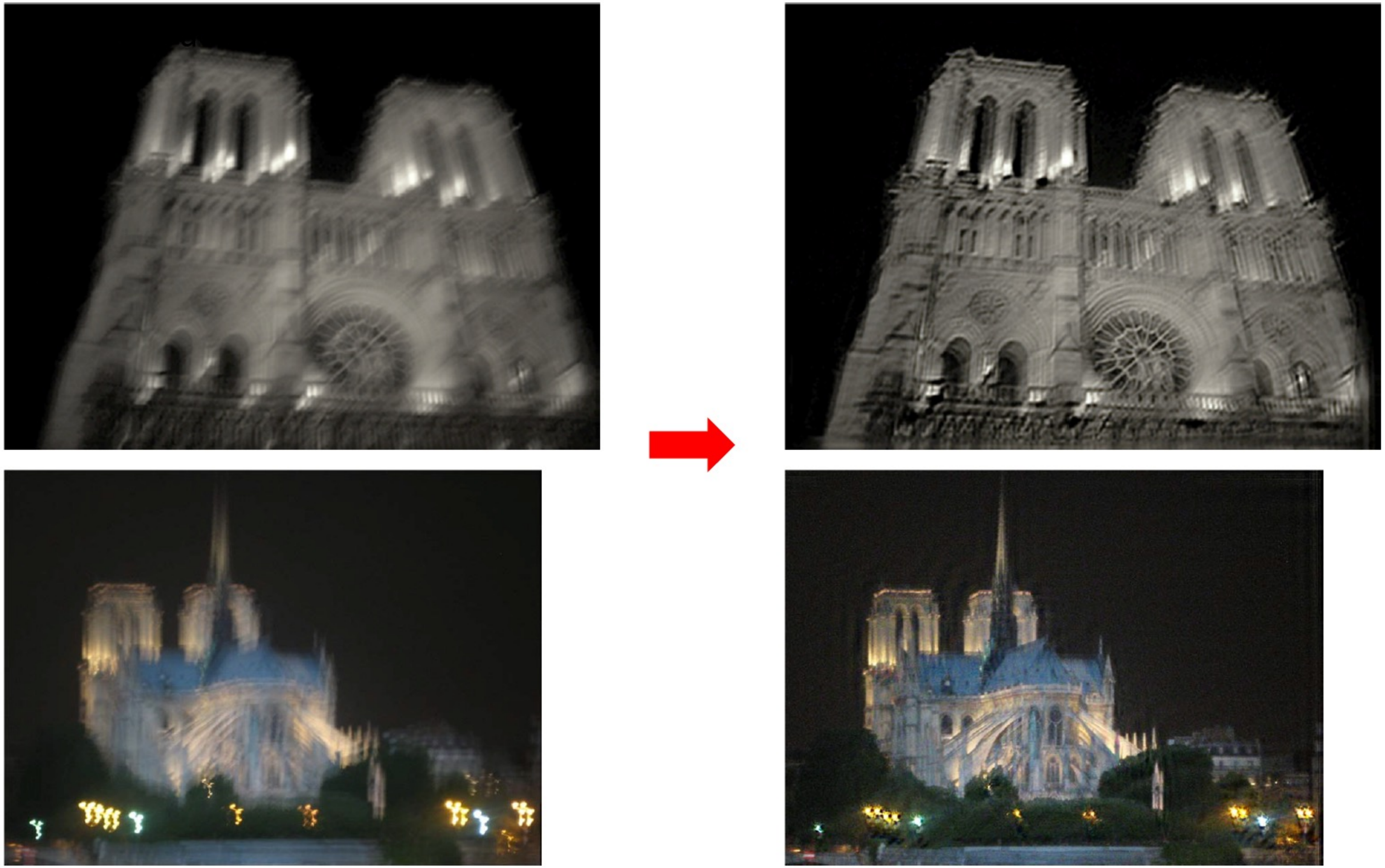
blurred image

estimated
blur filter



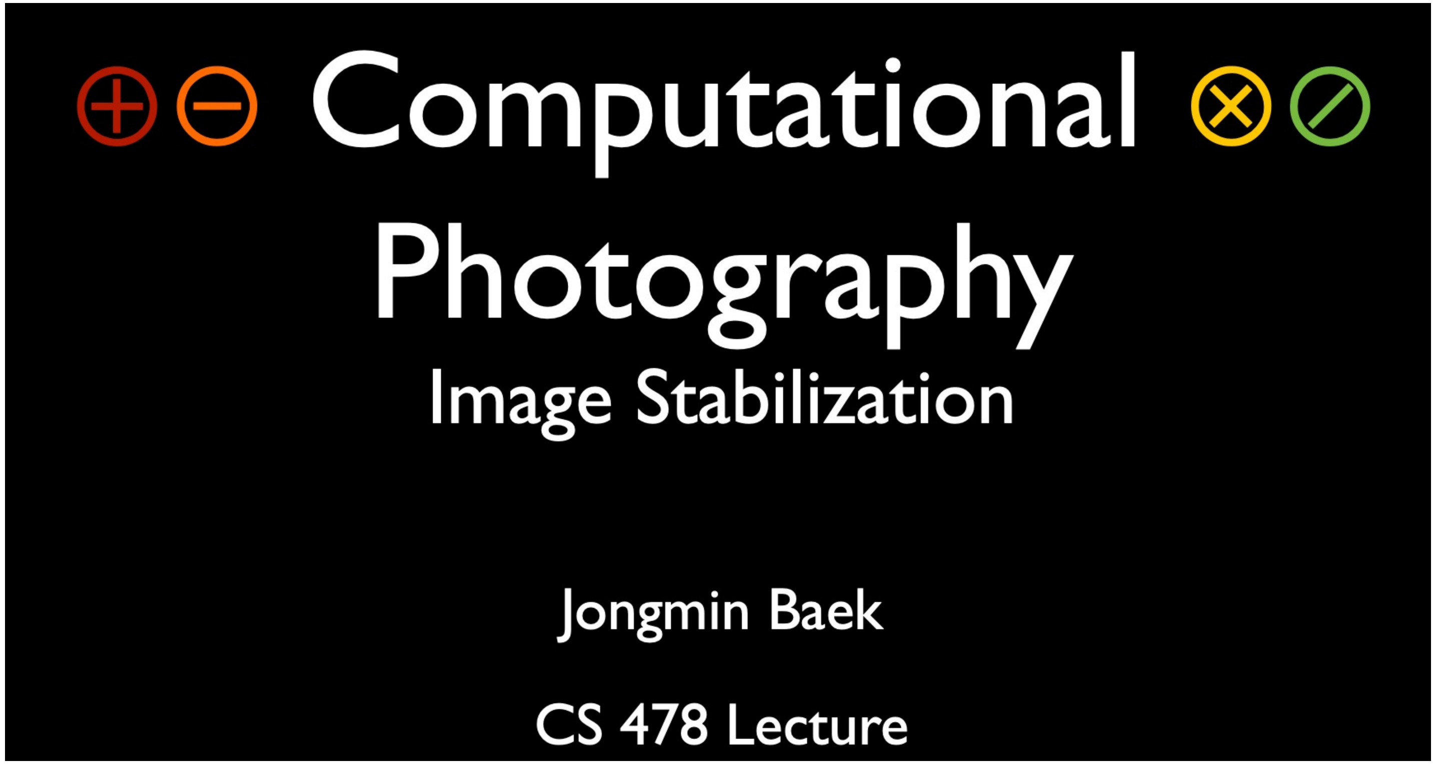
restored image

# More examples of blind deblurring



# Camera Shake Removal

- The next stack of slides are borrowed from:



# Blurs in Photography

- Defocus Blur

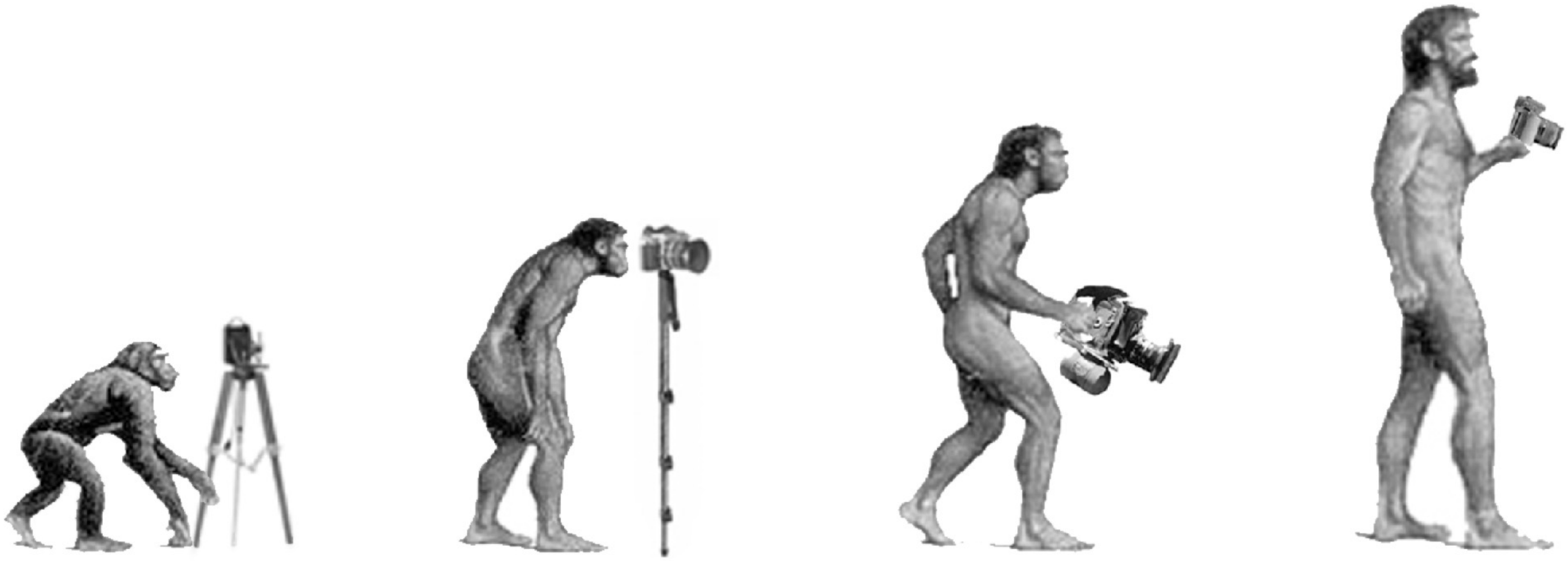
1/60 sec, f/1.8, ISO 400

# Blurs in Photography

- Handshake

2 sec, f/10, ISO 100

# Remove camera shake: Camera motion deblur



Tripod Monopod Heavy Weight Image Stabilizer

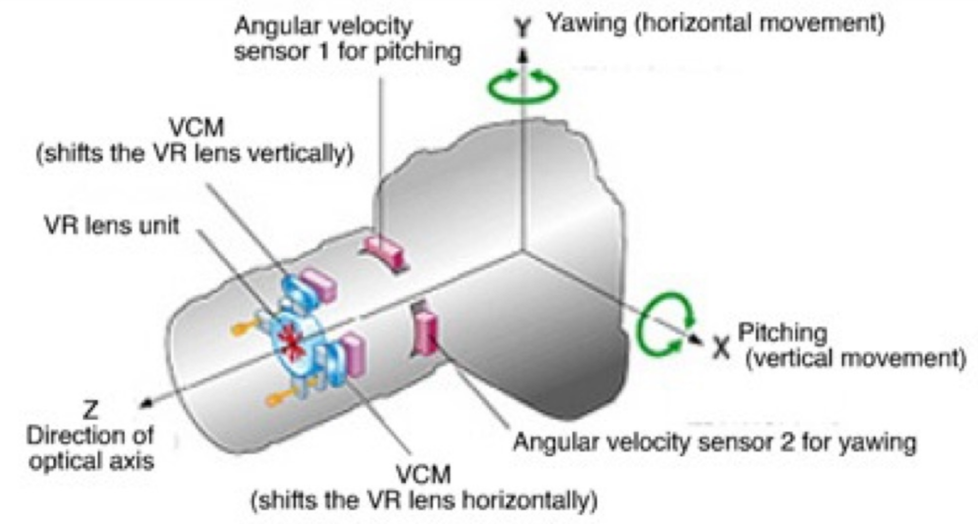
# Optical Image Stabilization

- Fight handshake.
  - Lens-Shift Image Stabilization
    - Vary the optical path to the sensor.
  - Sensor-Shift Image Stabilization
    - Move the sensor to counteract motion.

# Lens-Shift
# Image Stabilization

- Lots of different names
  - Image Stabilization (Canon)
  - Vibration Reduction (Nikon)
  - Optical Stabilization (Sigma)
  - Vibration Compensation (Tamron)
  - Mega OIS (Panasonic, Leika)

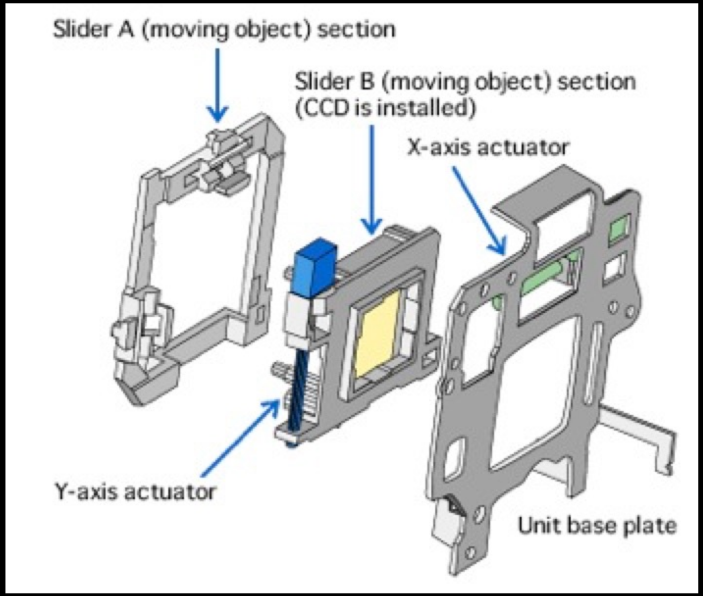
# Lens-Shift Image Stabilization




- A floating lens element moves orthogonally to the optical axis, using electromagnets.
  - Vibration is detected by two gyroscopes.
  - Pitch and yaw movements are compensated.
  - Roll and linear movement are not.

figures stolen from Sung Hee Park

# Sensor-Shift Image Stabilization


figures stolen from
Sung Hee Park

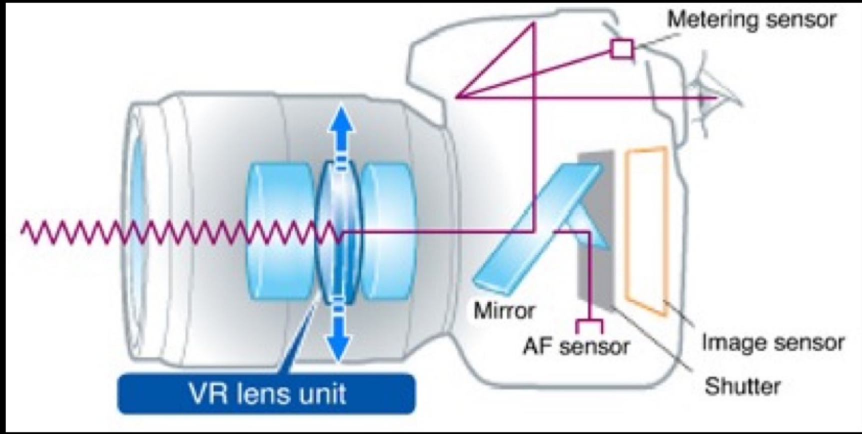
# Lens-Shift vs. Sensor-Shift

## Lens-Shift

- Stable viewfinder
- Better AF/AW
- Optimized to every lens

## Sensor-Shift

- Works for all lens
- Cost-effective
- Better optical performance




figures stolen from Sung Hee Park

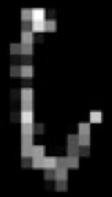
# Digital Stabilization

- What if you already incurred blur?
  - Need to “remove” blur

# Image Formation

- $I = L \otimes K + N$
  - $I$ : Observation
  - $L$ : Latent image
  - $K$ : Blur kernel
  - $N$ : Noise

L



K

N

I

# Some Results

# Some Results

Now let us read a few classic papers on digital camera shake removal…

# 1st Paper to read today (SIGGRAPH):

## Removing Camera Shake from a Single Photograph

Rob Fergus[1] Barun Singh[1] Aaron Hertzmann[2] Sam T. Roweis[2] William T. Freeman[1]

[1]MIT CSAIL [2]University of Toronto

Figure 1: *Left*: An image spoiled by camera shake. *Middle*: result from Photoshop "unsharp mask". *Right*: result from our algorithm.

# 2nd Paper to read today  (SIGGRAPH):

*To appear in the ACM SIGGRAPH conference proceedings*

## Image Deblurring using Inertial Measurement Sensors

Neel Joshi     Sing Bing Kang     C. Lawrence Zitnick     Richard Szeliski
Microsoft Research

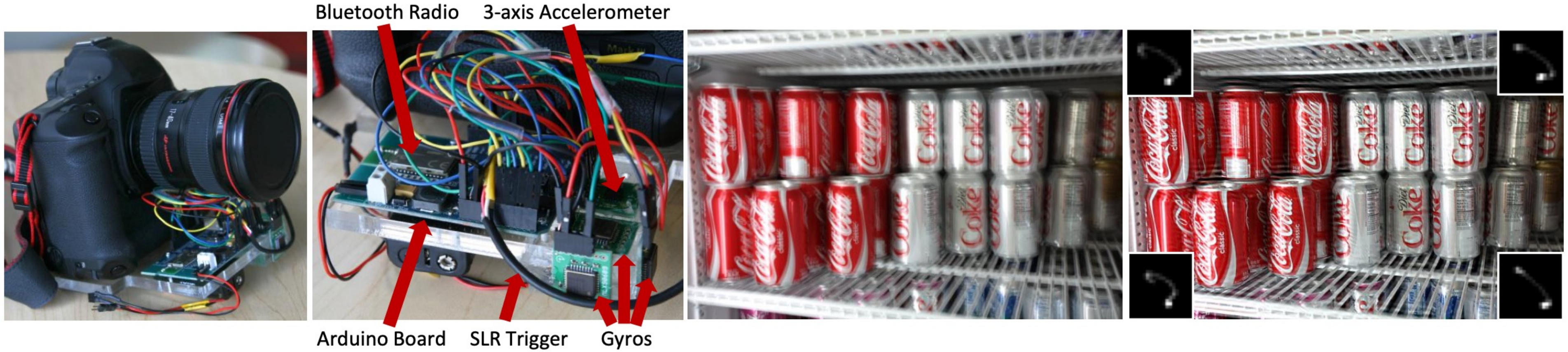



**Figure 1:** *An SLR Camera instrumented with our image deblurring attachment that uses inertial measurement sensors and the input image in an "aided blind-deconvolution" algorithm to automatically deblur images with spatially-varying blurs (first two images). A blurry input image (third image) and the result of our method (fourth image). The blur kernel at each corner of the image is shown at 2× size.*

# 3rd Paper to read today (CVPR16):

## Image Deblurring using Smartphone Inertial Sensors

Zhe Hu[1], Lu Yuan[2], Stephen Lin[2], Ming-Hsuan Yang[1]
[1]University of California, Merced [2]Microsoft Research
https://eng.ucmerced.edu/people/zhu/CVPR16_sensordeblur.html/

### Abstract

*Removing image blur caused by camera shake is an*

those derived from natural image statistics), the distribution of camera motions is difficult to model due to its large diversity and inherent randomness. To gain additional informa-

# Guided paper reading:

# Format:

1. Give you 5 minutes (break time) to refresh your memory of pre-lecture self-reading ...
2. Also, check your name on ZOOM is correct → first name + last name.
3. Check your mic and camera (if suitable).

4. **Supervised paper reading session: 30 minutes.**

5. Breakout room: 15 minutes. Unmute yourself, turn on the camera if suitable, critical analysis questions to discussion, Nominate a "Speaker", take notes (record your room-ID, and all the student names, speaker's name, and key point of discussion) or write on PPT slides,
6. 5x random groups; report back, 15 minutes. Share screen, using PPT, Each group 3 minutes.

- Switch to PDF paper reading … 30 minutes.

# Breakout room: Now it is your turn.

- Breakout room  group discussion & report:
- Check your name on Zoom
- Take note: breakout room ID, room members, “Mr/Ms speaker”.

- Your task:
  - Pick one paper, either (1) the ”Rob Fergus”,  or  (2 + 3) “Neel Joshi” and “Zhe Hu”.
  - Discuss their limitations/drawbacks.

  - Try to answer the following “critical analysis” questions → page turn ..
  - Report back in 3 minutes by the “Mr/Ms speaker”.  Tutor may record your report.

# Critical Analysis

### 5.1. Are the paper's contributions significant?

Are the contribution/improvement trivial, incremental?
Why previous efforts failed ?

### 5.2. Are the authors' main claims valid?

Have they convincingly validated their main idea?
Any hole in their arguments, derivation, experiments ?

### 5.3. Limitation and weaknesses

Any limitation/weakness of their method? What can be done to improve the work ?

What would you do to address/overcome their weaknesses?

### 5.4. Extension and future work

What *extra* experiments that you would suggest the authors to conduct in order to strengthen the result ?

Can you think of other possible applications of the method/ideas (assuming valid) presented in the paper?

What are possible future works?

### 5.5. Is the paper stimulating or inspiring ?

Many papers (even those published ones) are dull, while some are exciting. What is your opinion about this paper and why?

### 5.6. Conclusion and personal reflection

First, draw a short conclusion about this paper.

Then, if you were tasked to solve the research problem, what would you do differently? an alternative solution ?

Finally, in one sentence, summarize what you have learn from reading this paper.

Some PPT slides for paper#1 borrowed from Fergus's SIGGRAPH talk.

# Removing Camera Shake from a Single Photograph

Rob Fergus[1] Barun Singh[1] Aaron Hertzmann[2] Sam T. Roweis[2] William T. Freeman[1]

[1]MIT CSAIL [2]University of Toronto

Figure 1: *Left*: An image spoiled by camera shake. *Middle*: result from Photoshop “unsharp mask”. *Right*: result from our algorithm.

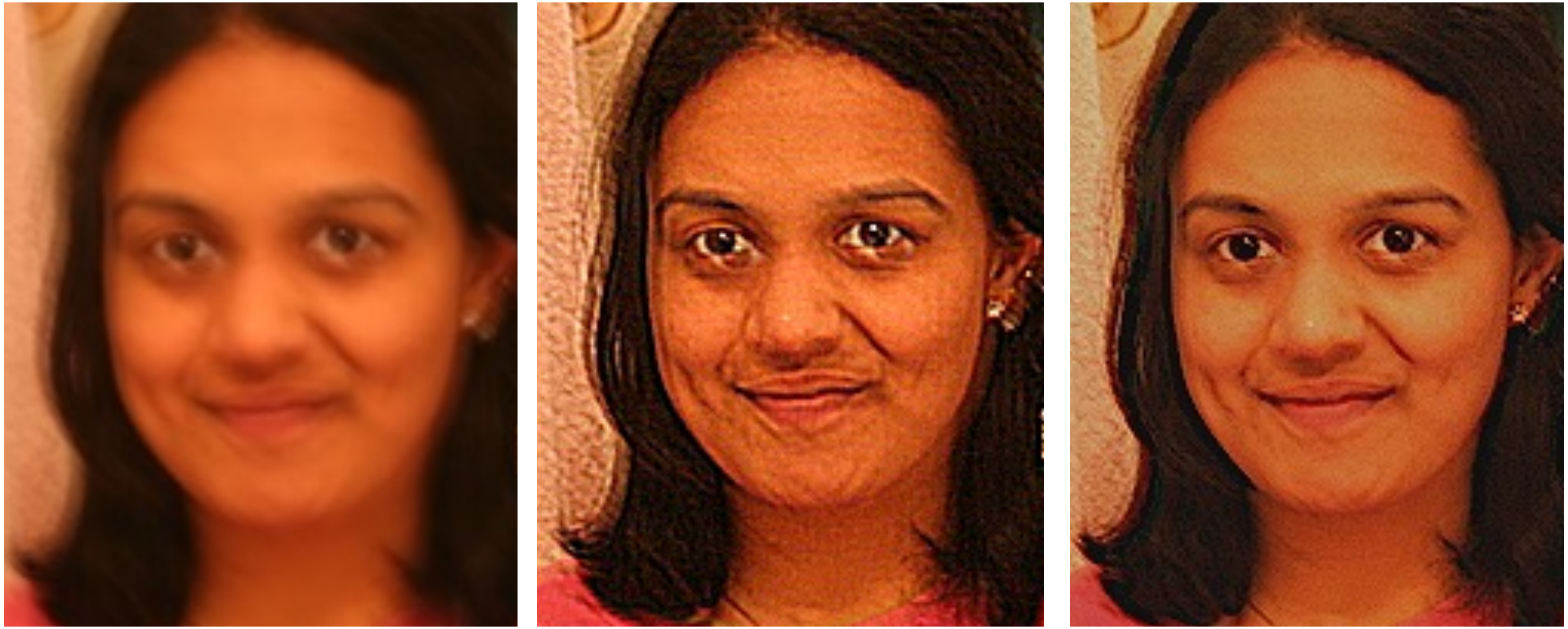
# Removing Camera Shake from a Single Photograph

Rob Fergus, Barun Singh, Aaron Hertzmann,
Sam T. Roweis and William T. Freeman

Massachusetts Institute of Technology
and
University of Toronto

# Overview

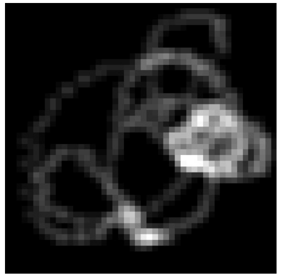
# Close-up



# Image formation process

=

$\otimes$



Blurry image

Sharp image

Blur kernel

Input to algorithm

Desired output

Convolution operator

Model is approximation

# Why is this hard?

Simple analogy:

11 is the product of two numbers.

What are they?

No unique solution:

11 = 1 x 11

11 = 2 x 5.5

11 = 3 x 3.667

etc…..

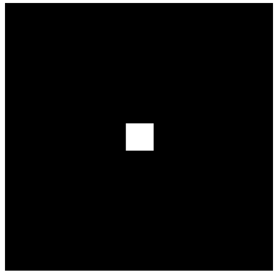
Need more information !!!!

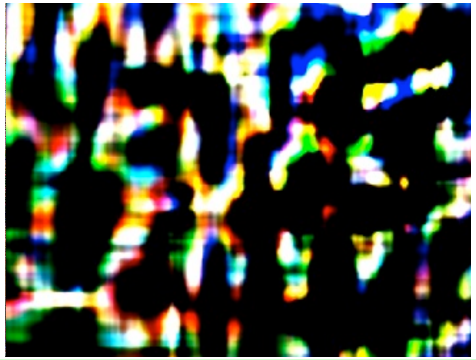
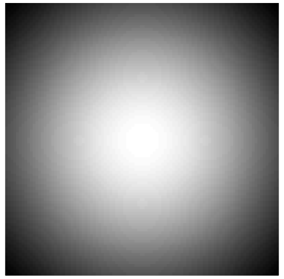
# Multiple possible solutions

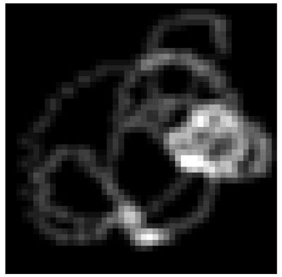
Sharp image

Blur kernel

Blurry image

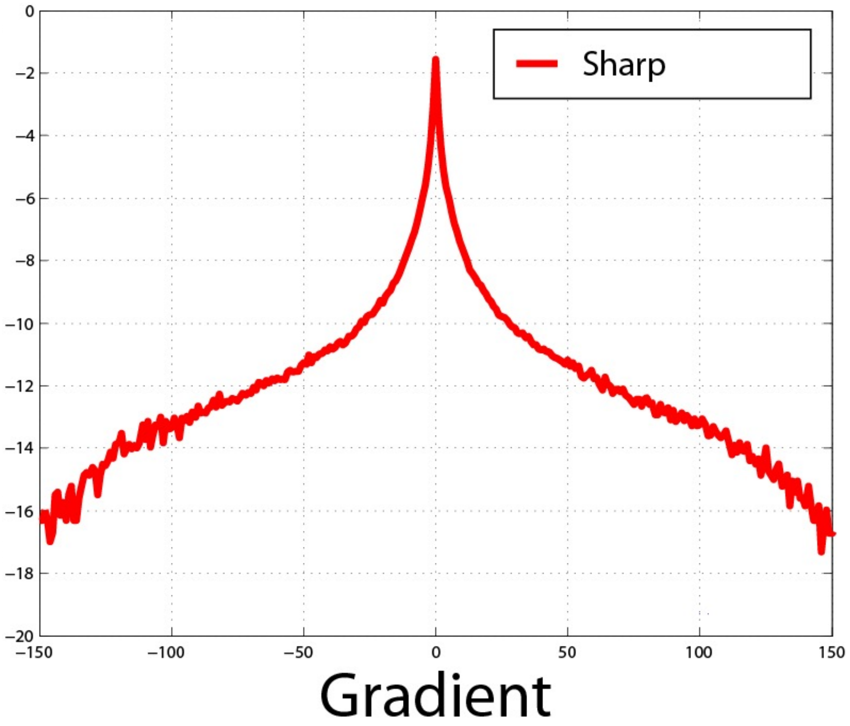
$=$  $\otimes$ 

$=$  $\otimes$ 

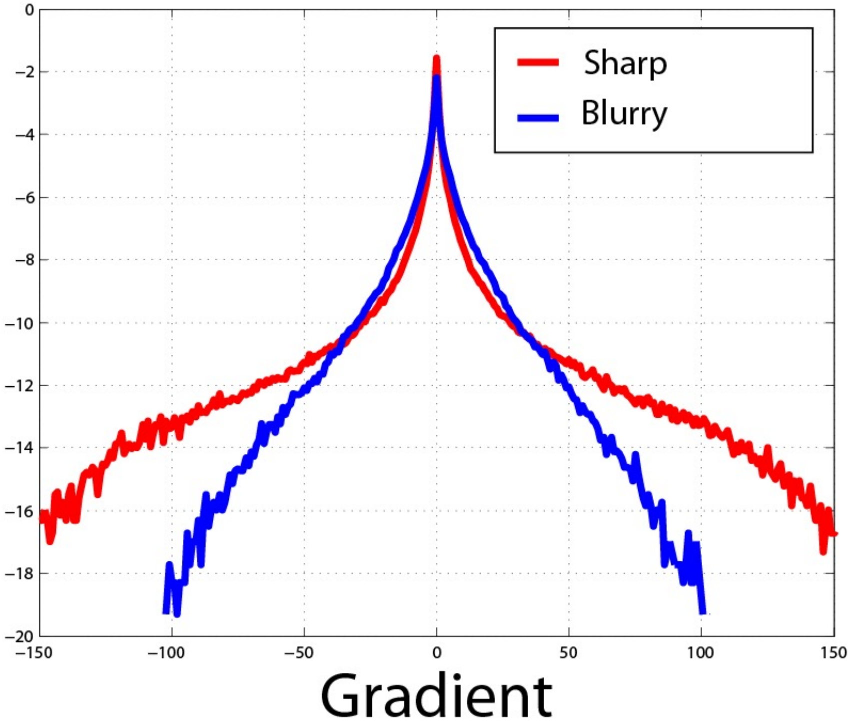
$=$  $\otimes$ 

# Natural image statistics

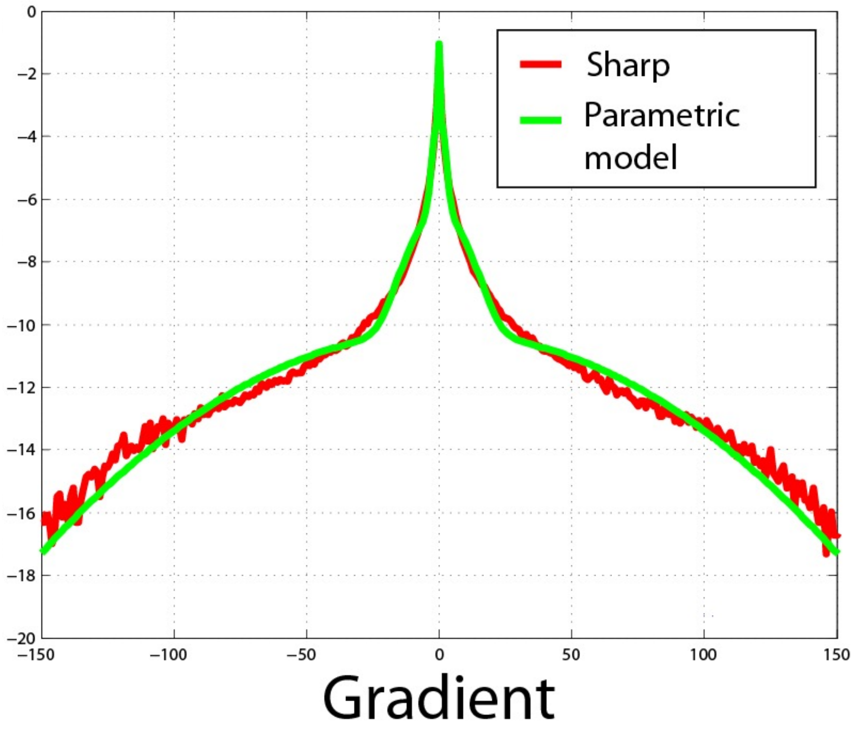
Characteristic distribution with heavy tails



# Blury images have different statistics



# Parametric distribution


Use parametric model of sharp image statistics

# Uses of natural image statistics

- Denoising [Roth and Black 2005]
- Superresolution [Tappen et al. 2005]
- Intrinsic images [Weiss 2001]
- Inpainting [Levin et al. 2003]
- Reflections [Levin and Weiss 2004]
- Video matting [Apostoloff & Fitzgibbon 2005]

Corruption process assumed known

# Existing work on image deblurring

## Software algorithms:

- Extensive literature in signal processing community

Mainly Fourier and/or Wavelet based

Strong assumptions about blur

→ not true for camera shake

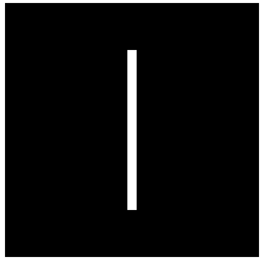
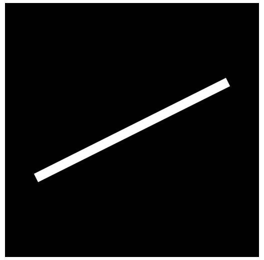
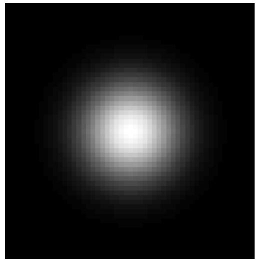
  

Image constraints are frequency-domain power-laws

# Three sources of information

## 1. Image formation constraint:

Estimated sharp image

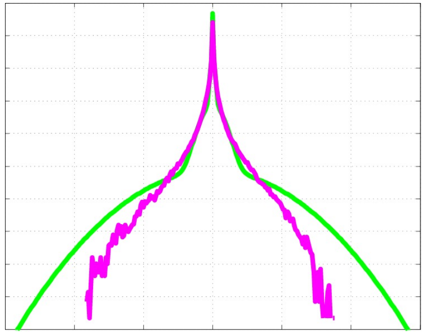
$\otimes$

Estimated blur kernel

=

Input blurry image

## 2. Image prior:



Distribution of gradients

## 3. Blur prior:

Positive & Sparse

# How do we use this information?

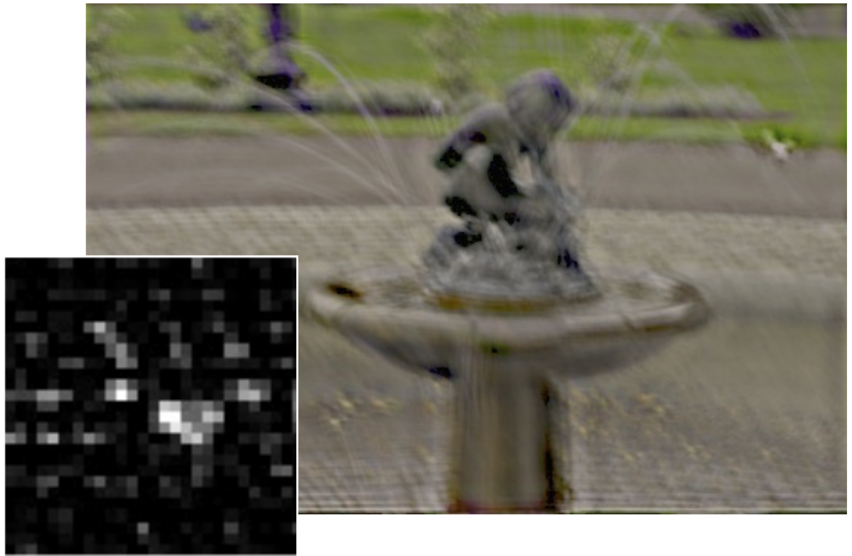
Obvious thing to do:

- Combine 3 terms into an objective function
- Run conjugate gradient descent
- This is Maximum a-Posteriori (MAP)

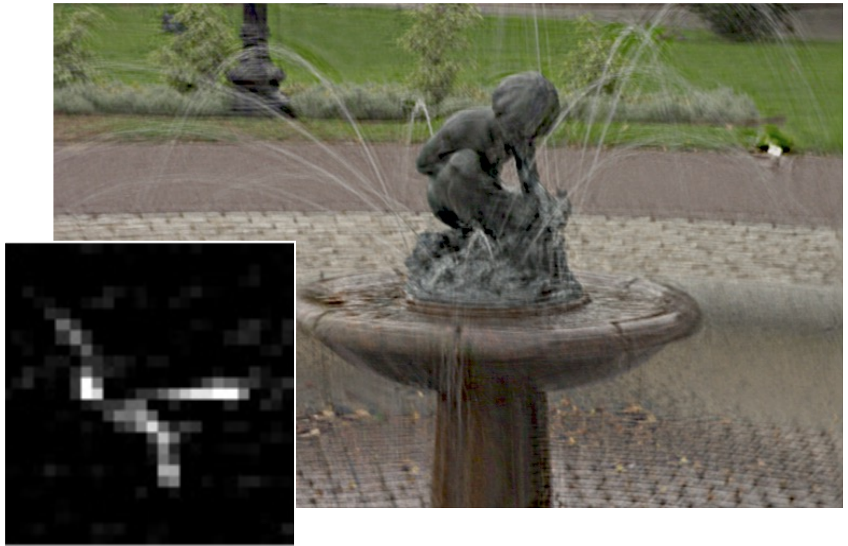
# Results from MAP estimation

Input blurry image

Maximum a-Posteriori (MAP)



Our method: Variational Bayes



# Overview of algorithm

1. Pre-processing

2. Kernel estimation
   - Multi-scale approach

3. Image reconstruction
   - Standard non-blind deconvolution routine

# Preprocessing

Input image

Convert to grayscale

Remove gamma correction

User selects patch from image

Bayesian inference too slow to run on whole image

Infer kernel from this patch

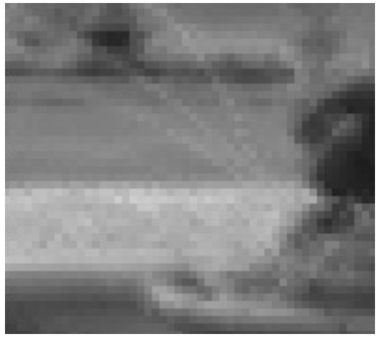
# Initialization

Input image

Convert to grayscale → Remove gamma correction → User selects patch from image

Initialize 3x3 blur kernel

Blurry patch



Initial image estimate

Initial blur kernel

# Inferring the kernel: multiscale method

Input image

Convert to grayscale

Remove gamma correction

User selects patch from image

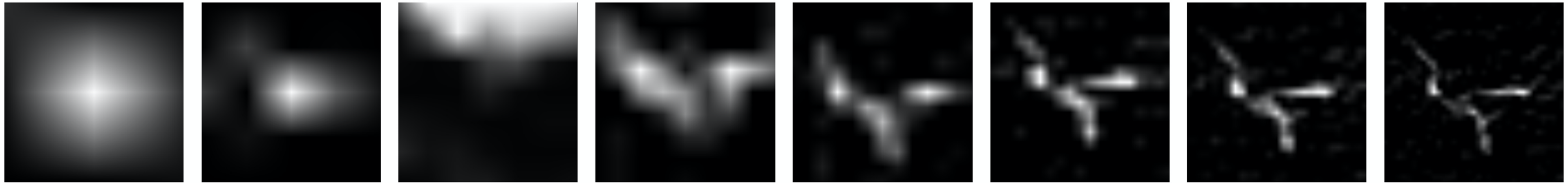
Initialize 3x3 blur kernel

Loop over scales

Variational Bayes

Upsample estimates

## Use multi-scale approach to avoid local minima:



# Image Reconstruction

Input image

Convert to grayscale

Remove gamma correction

User selects patch from image

Initialize 3x3 blur kernel

Variational Bayes

Loop over scales

Upsample estimates

Full resolution blur estimate

Non-blind deconvolution (Richardson-Lucy)

Deblurred image

# Results on real images

Submitted by people from their own photo collections

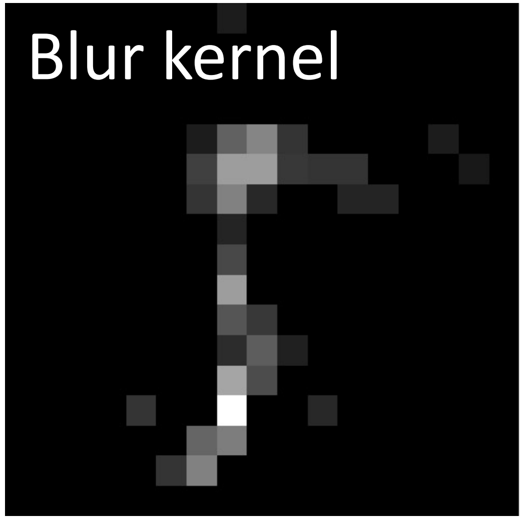
Type of camera unknown

Output does contain artifacts

- Increased noise
- Ringing

Compares well to existing methods

# Original photograph


Blur kernel


Our output

# Matlab's `deconvblind`

# Close-up of garland

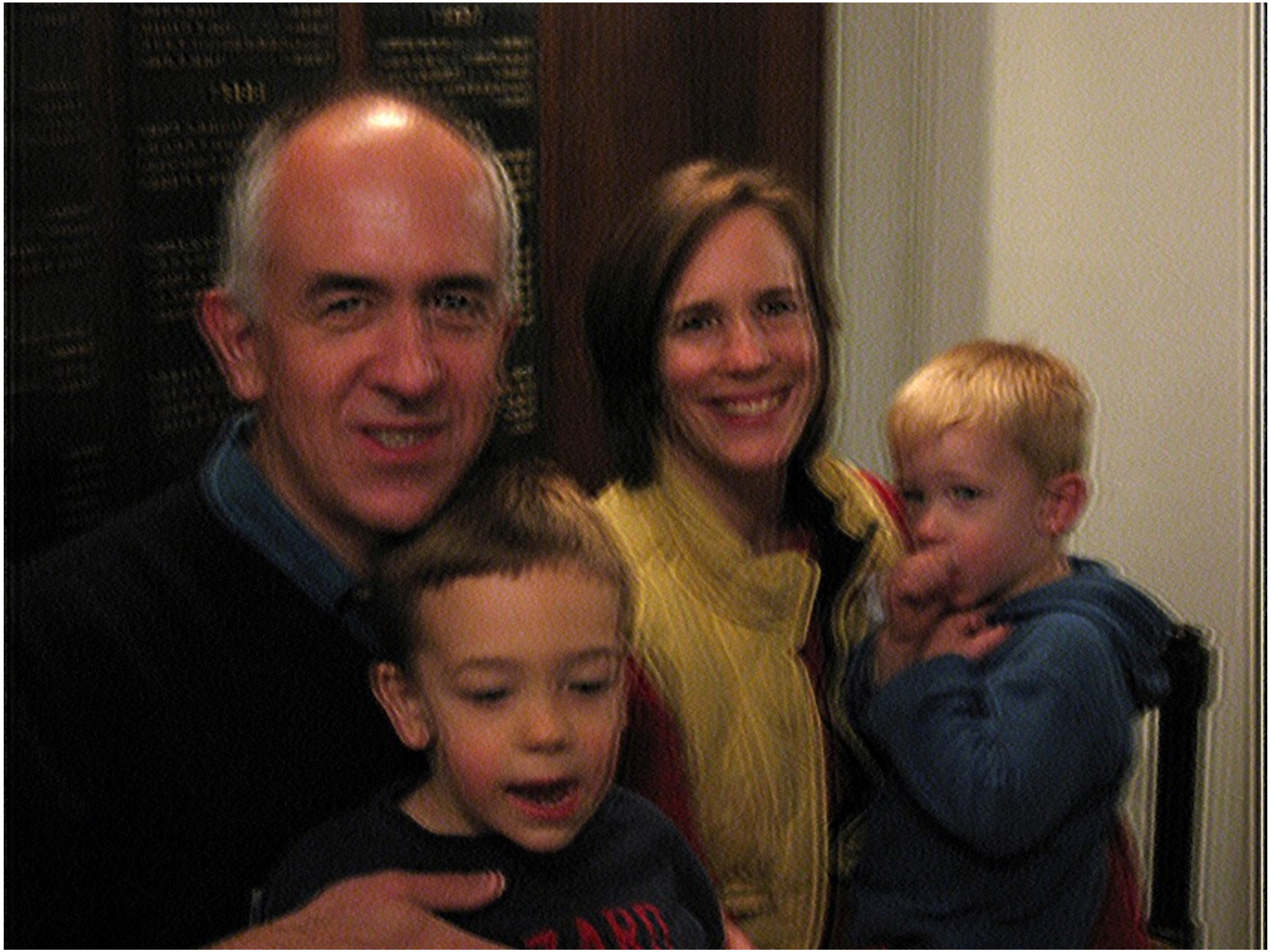
Original

Matlab's
`deconvblind`

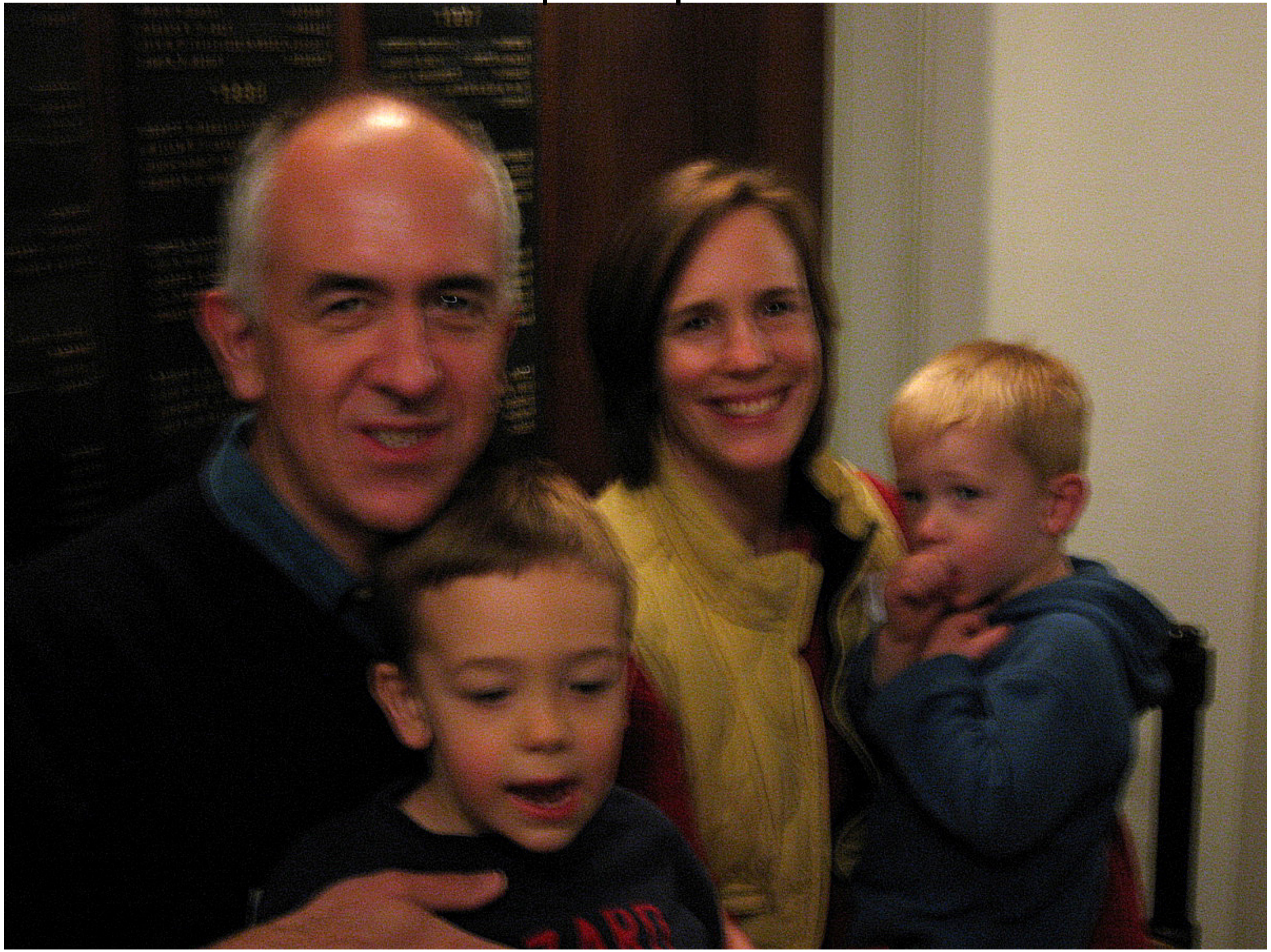
Our output

# Original photograph

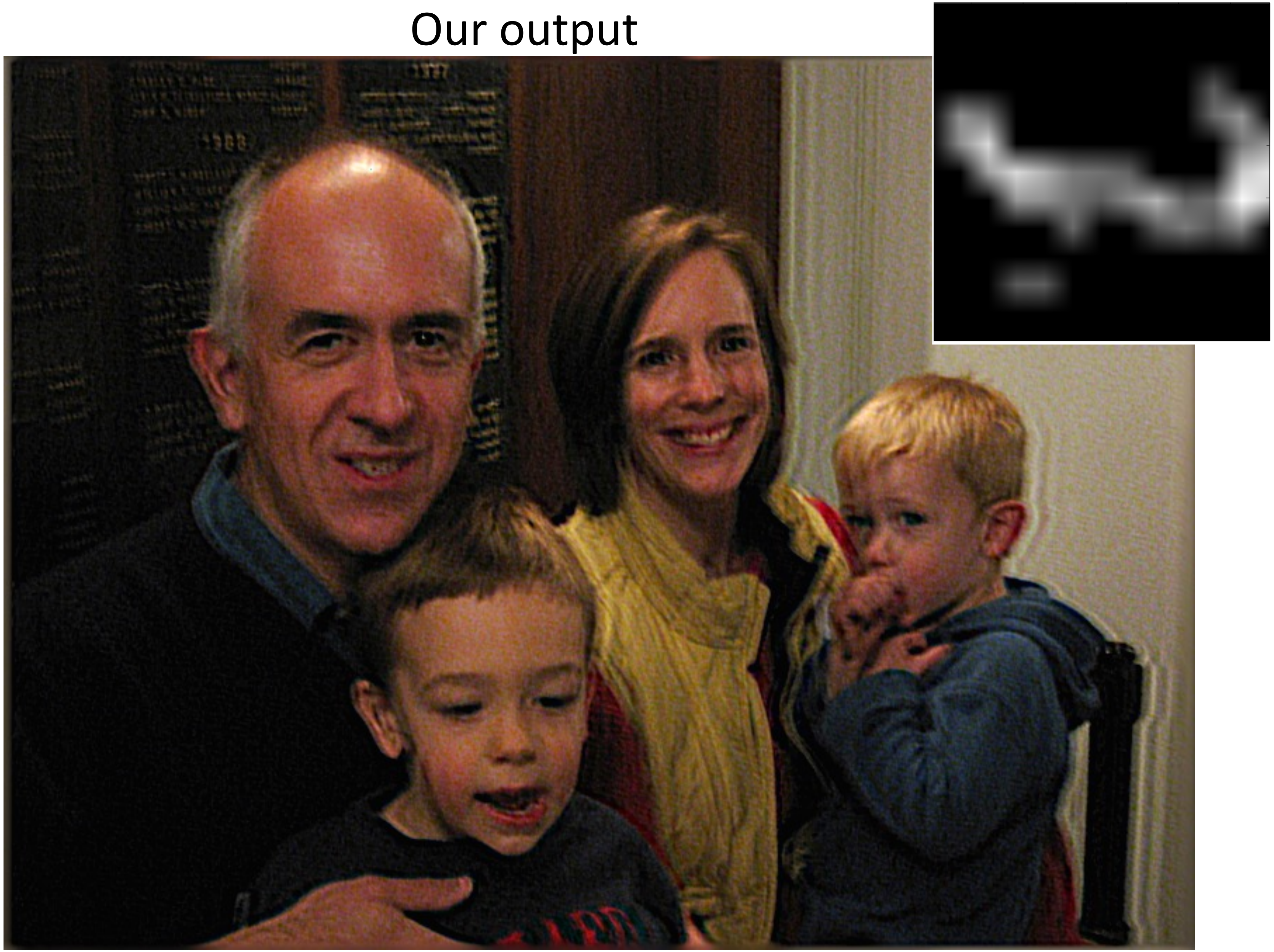
# Matlab's `deconvblind`

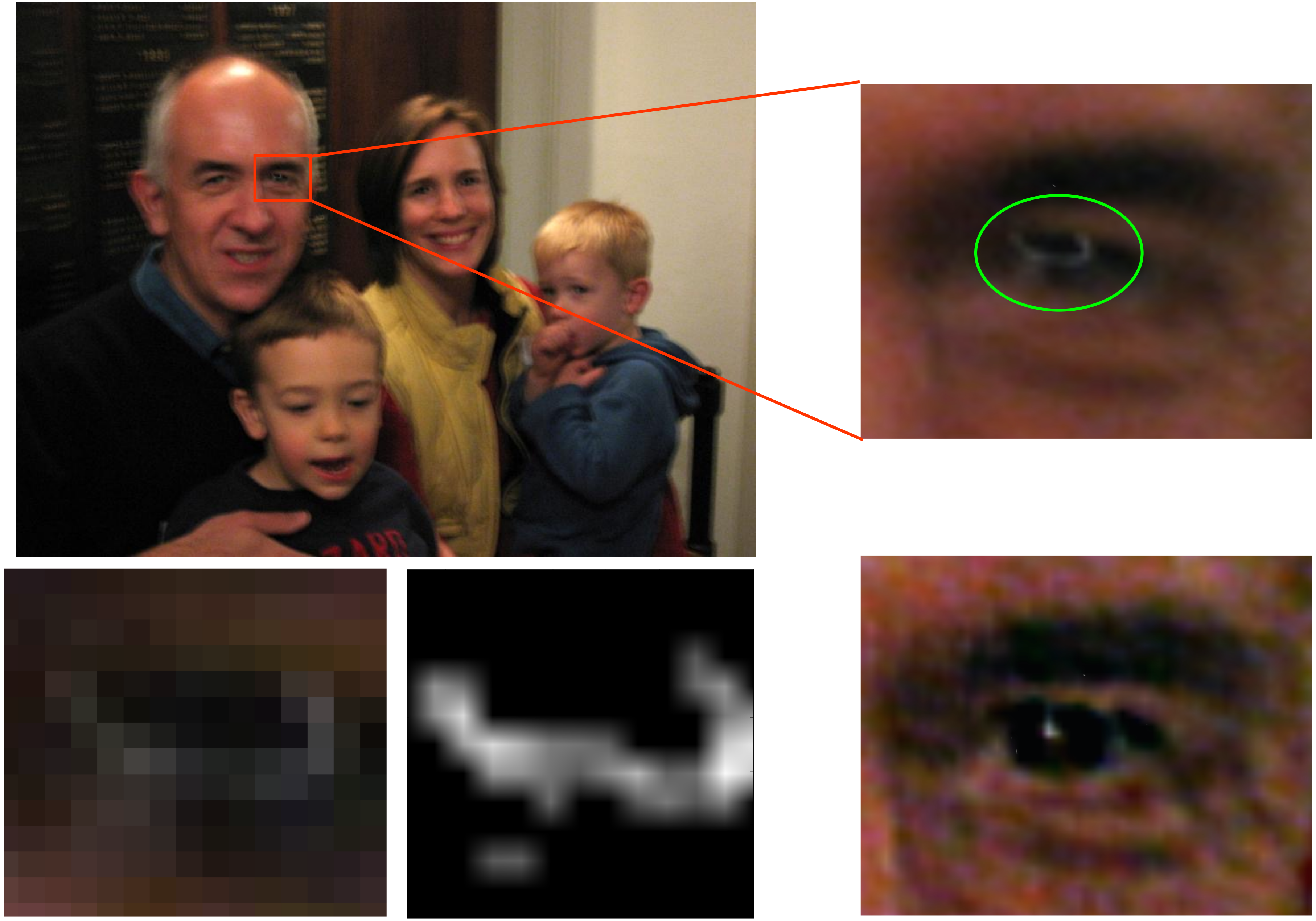


# Photoshop sharpen more



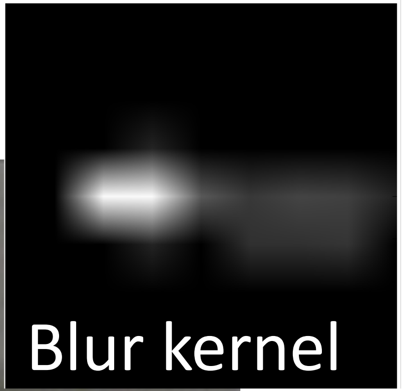
# Our output

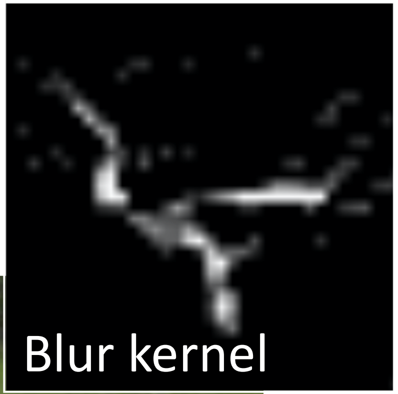


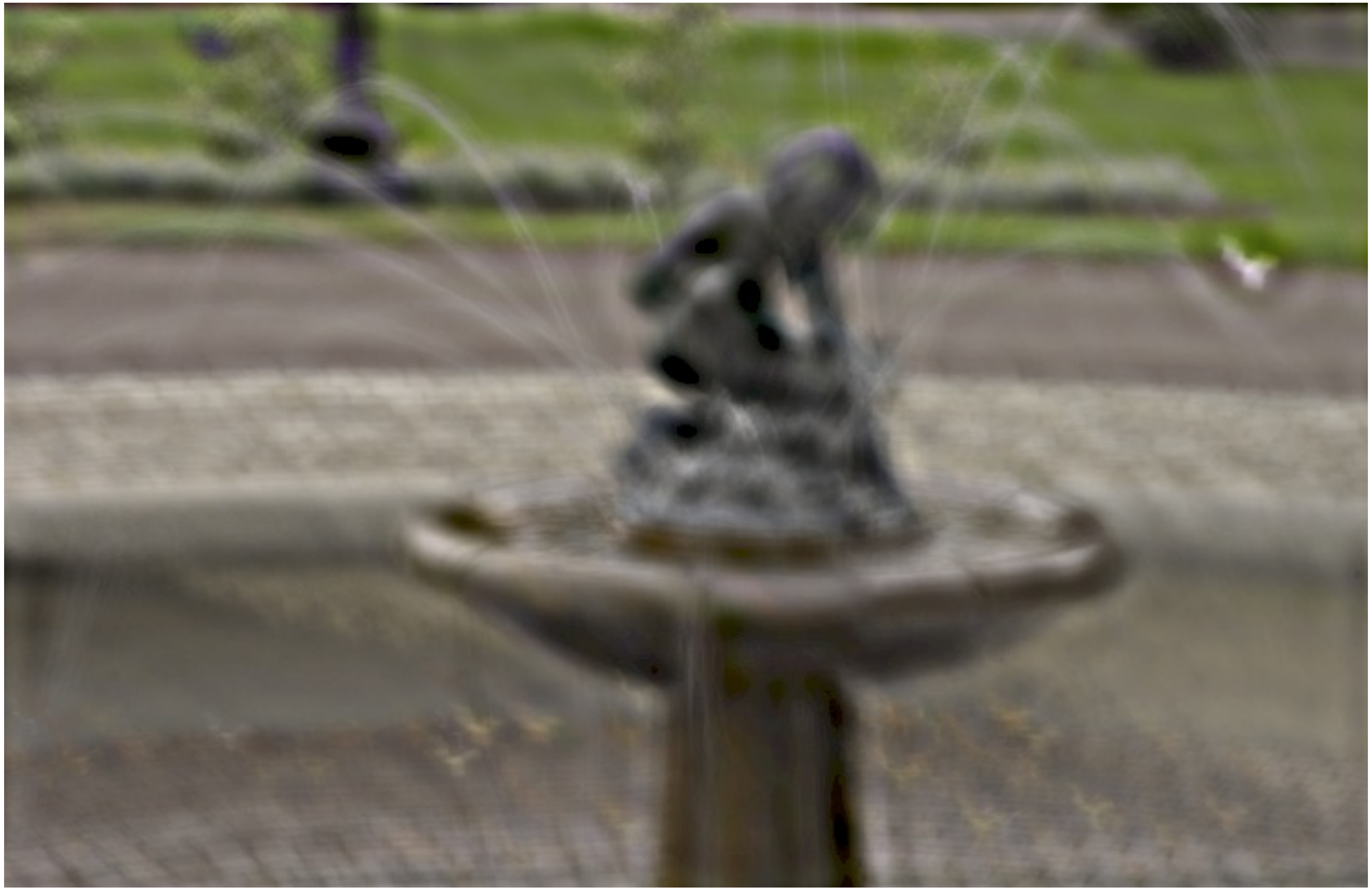


# Original photograph

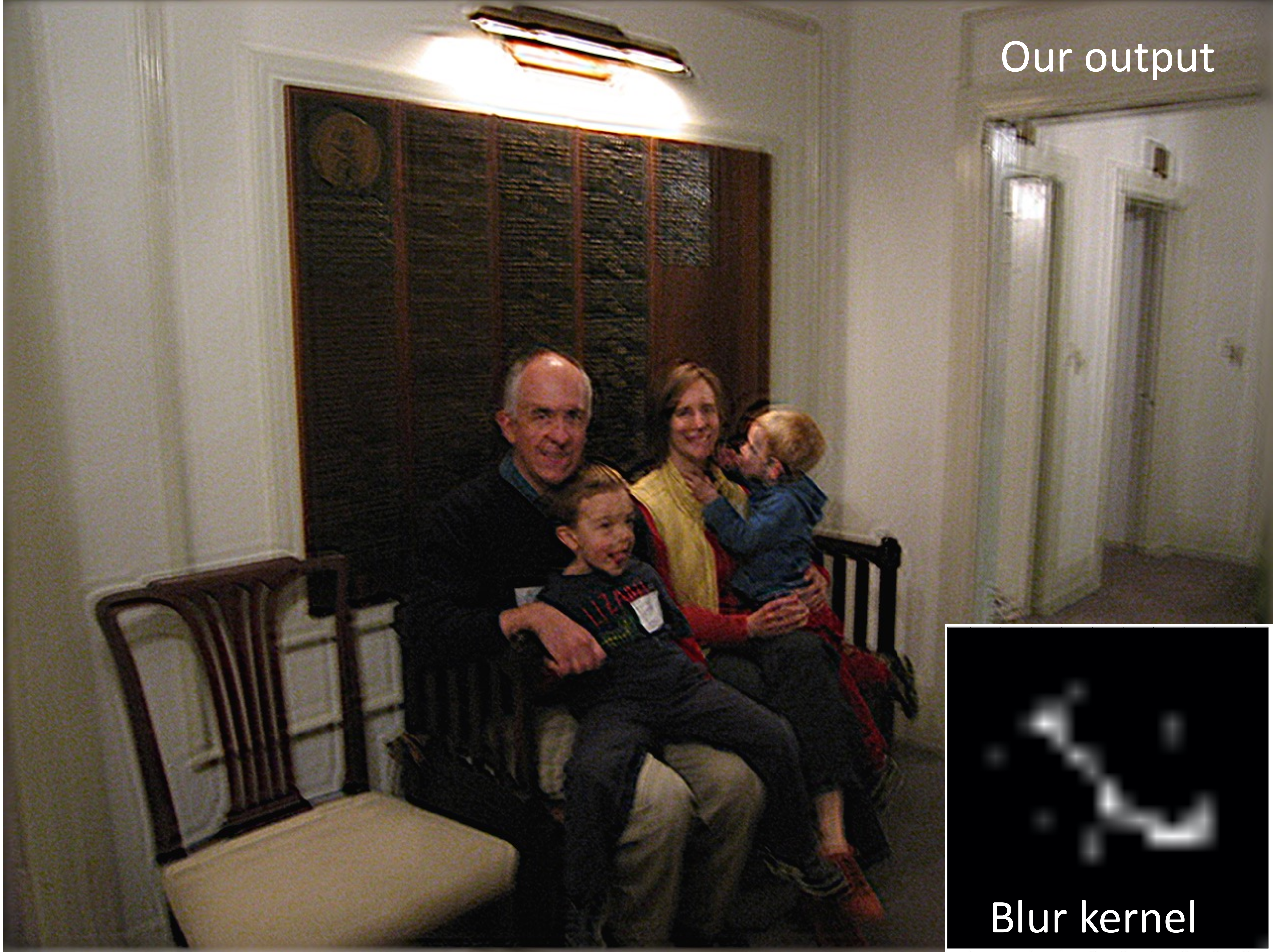
Our output

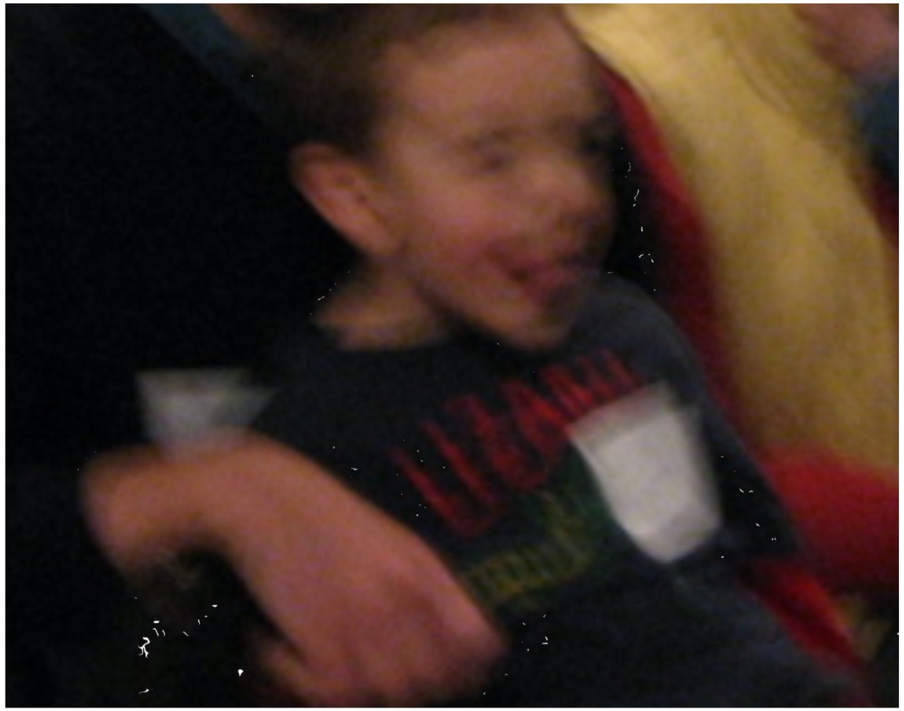

Blur kernel



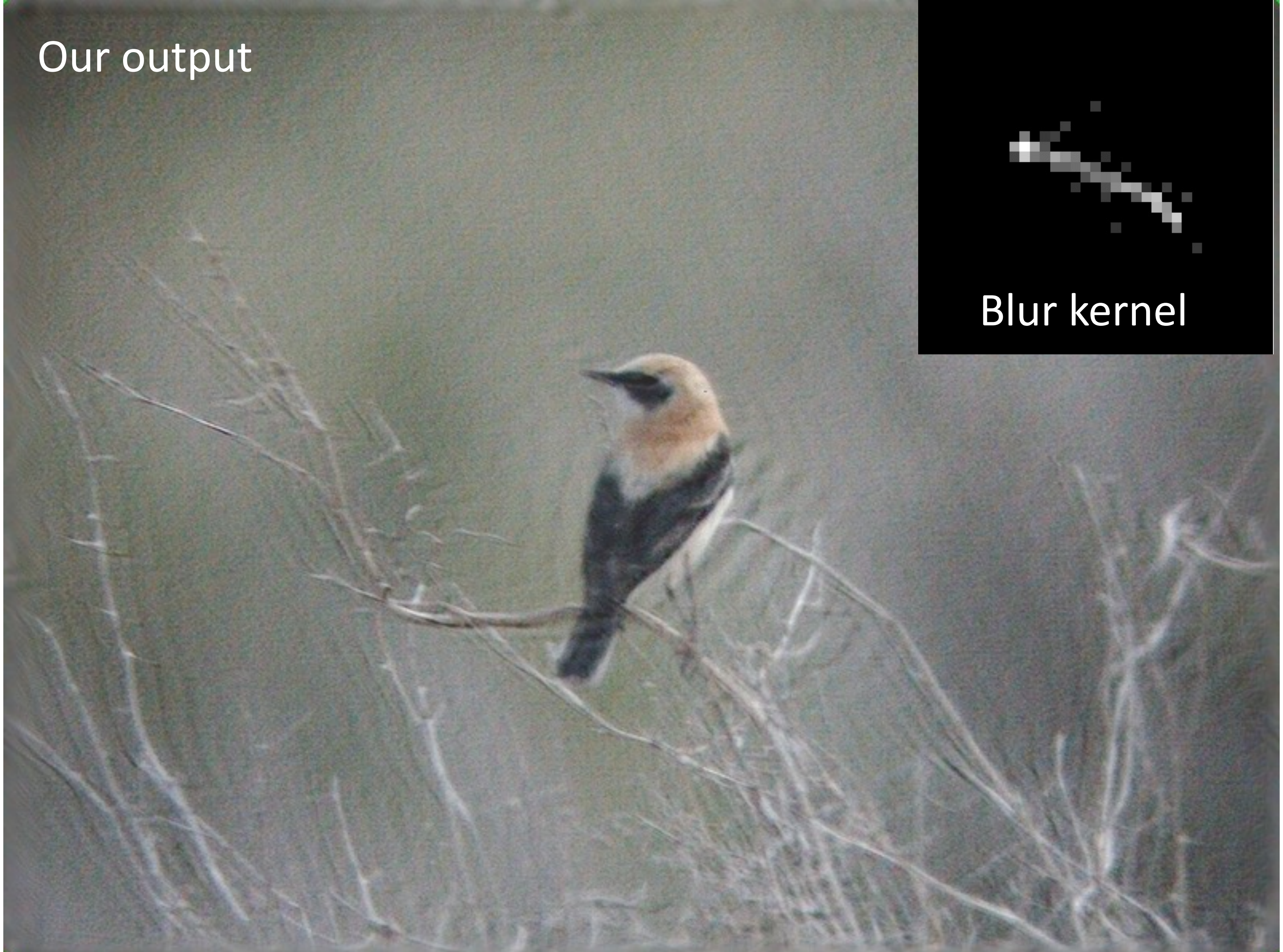
Original photograph


Our output
Blur kernel

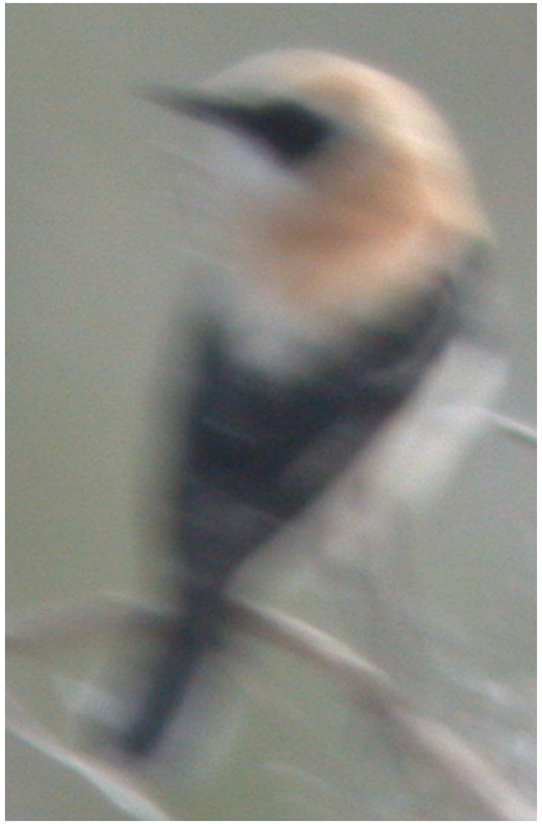
# Close-up of child




Original photograph



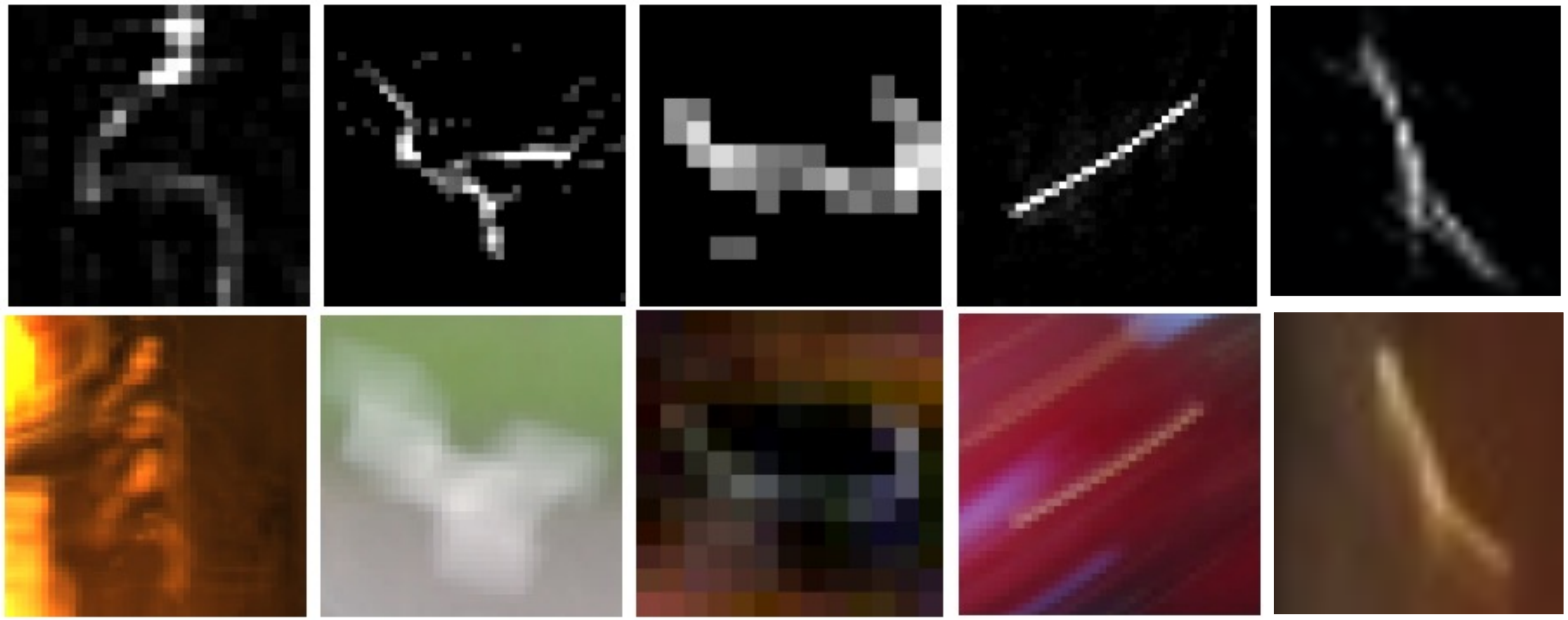
Our output
Blur kernel




Original photograph


Our output

# Image artifacts & estimated kernels



Note: blur kernels were inferred from large image patches,
NOT the image patterns shown


Blurry synthetic image

Our output

Ground truth

Matlab's `deconvblind`


# Summary

Method for removing camera shake
from real photographs

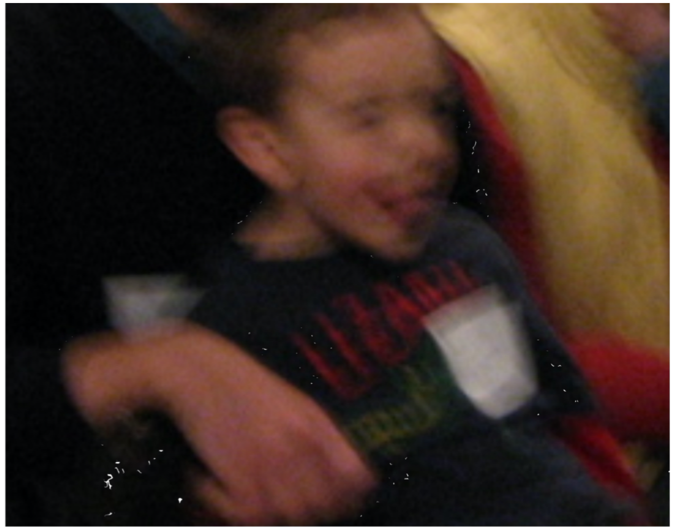
First method that can handle
complicated blur kernels

Uses natural image statistics

Non-blind deconvolution
currently simplistic

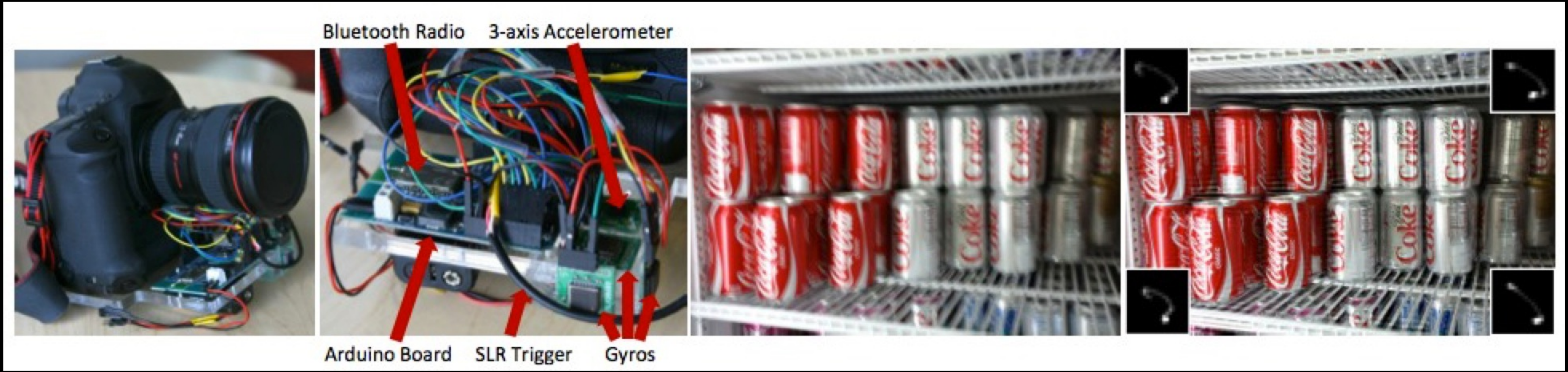
Things we have yet to model:

- Correlations in colors, scales, kernel continuity
- JPEG noise, saturation, object motion



# Other Twists

- Use gyros to figure out kernel



# 2nd Paper to read today  (SIGGRAPH):

*To appear in the ACM SIGGRAPH conference proceedings*

## Image Deblurring using Inertial Measurement Sensors

Neel Joshi     Sing Bing Kang     C. Lawrence Zitnick     Richard Szeliski
Microsoft Research

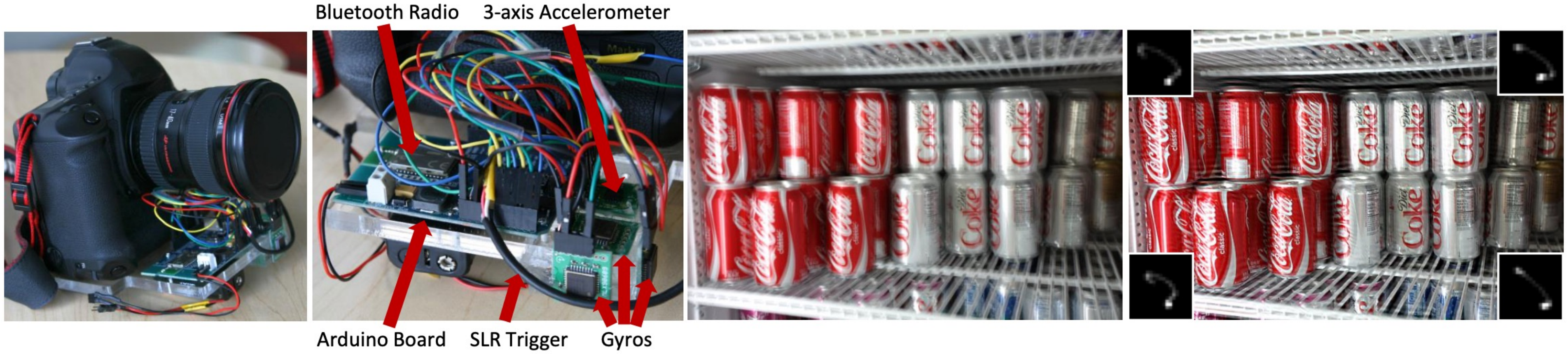



**Figure 1:** *An SLR Camera instrumented with our image deblurring attachment that uses inertial measurement sensors and the input image in an "aided blind-deconvolution" algorithm to automatically deblur images with spatially-varying blurs (first two images). A blurry input image (third image) and the result of our method (fourth image). The blur kernel at each corner of the image is shown at 2× size.*

# Camera rotational motion

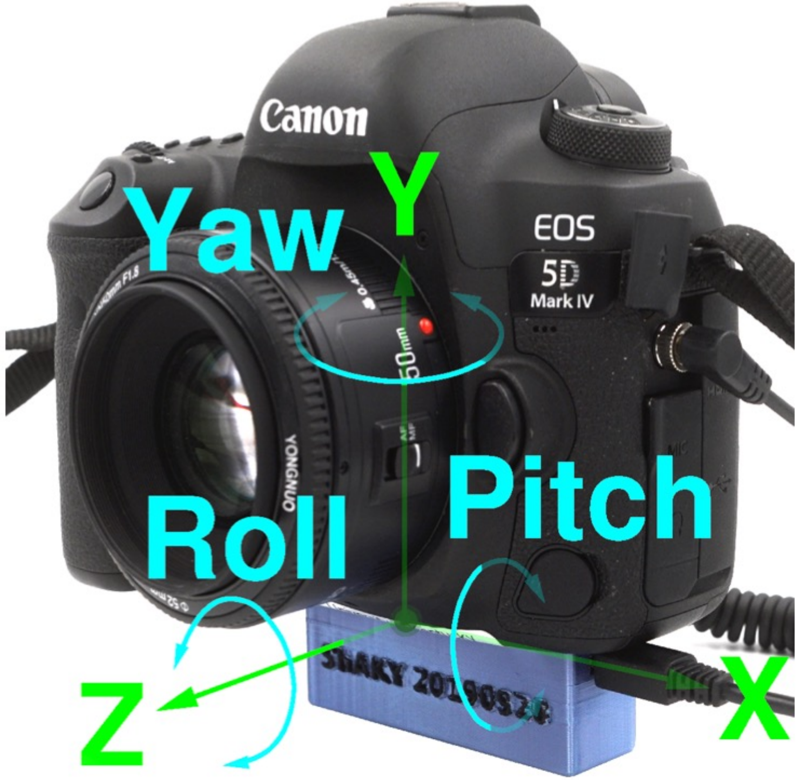
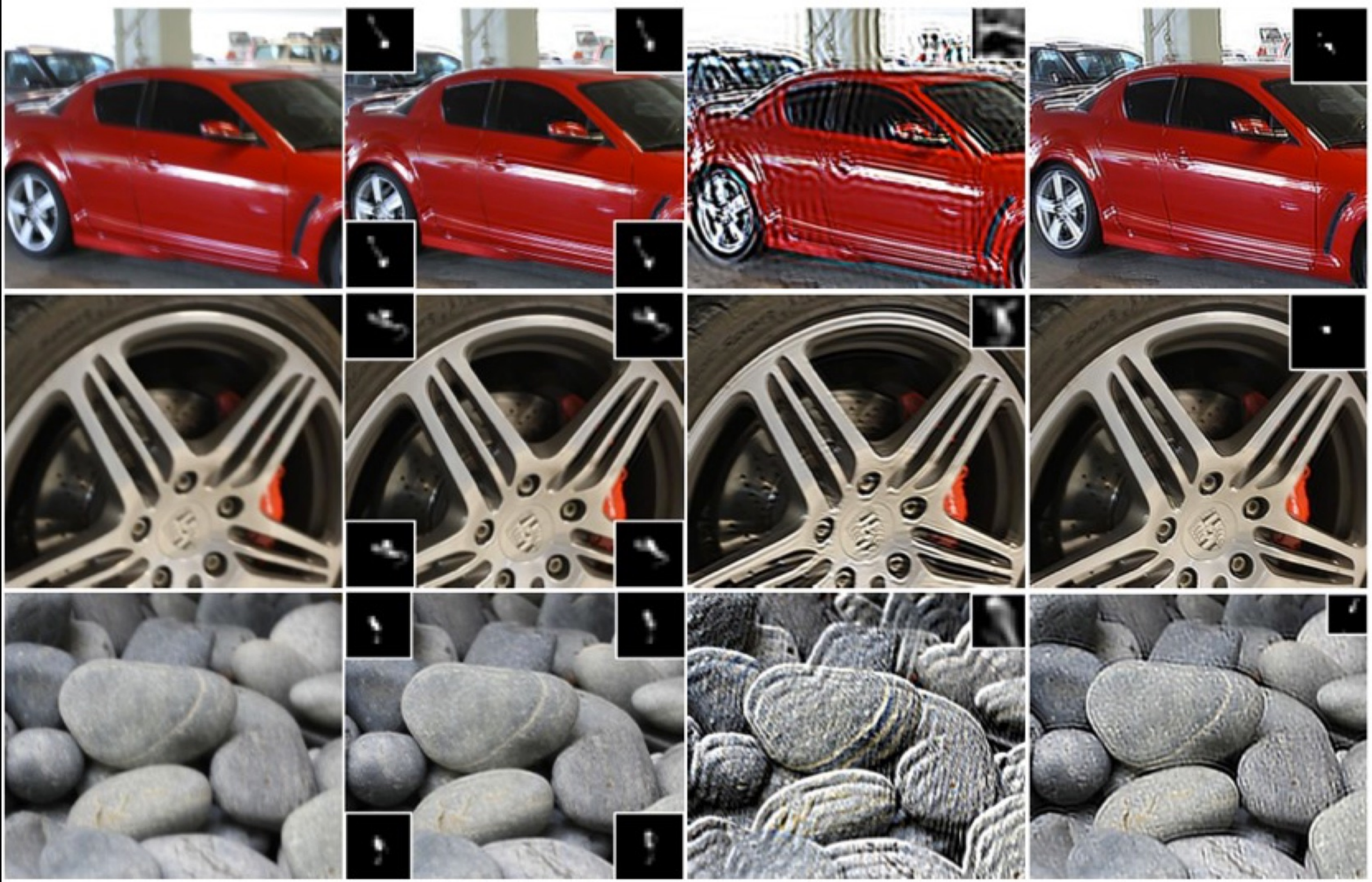



***Figure 1.*** *Measurement orientations for ShAKY on a Canon 5DIV*

# Other Twists



Input | IMU Deblurring (Our Final Output) | Shan et al. | Fergus et al.

# 3rd Paper to read today (CVPR16):

## Image Deblurring using Smartphone Inertial Sensors

Zhe Hu[1], Lu Yuan[2], Stephen Lin[2], Ming-Hsuan Yang[1]
[1]University of California, Merced [2]Microsoft Research
https://eng.ucmerced.edu/people/zhu/CVPR16_sensordeblur.html/

### Abstract

*Removing image blur caused by camera shake is an*

those derived from natural image statistics), the distribution of camera motions is difficult to model due to its large diversity and inherent randomness. To gain additional informa-

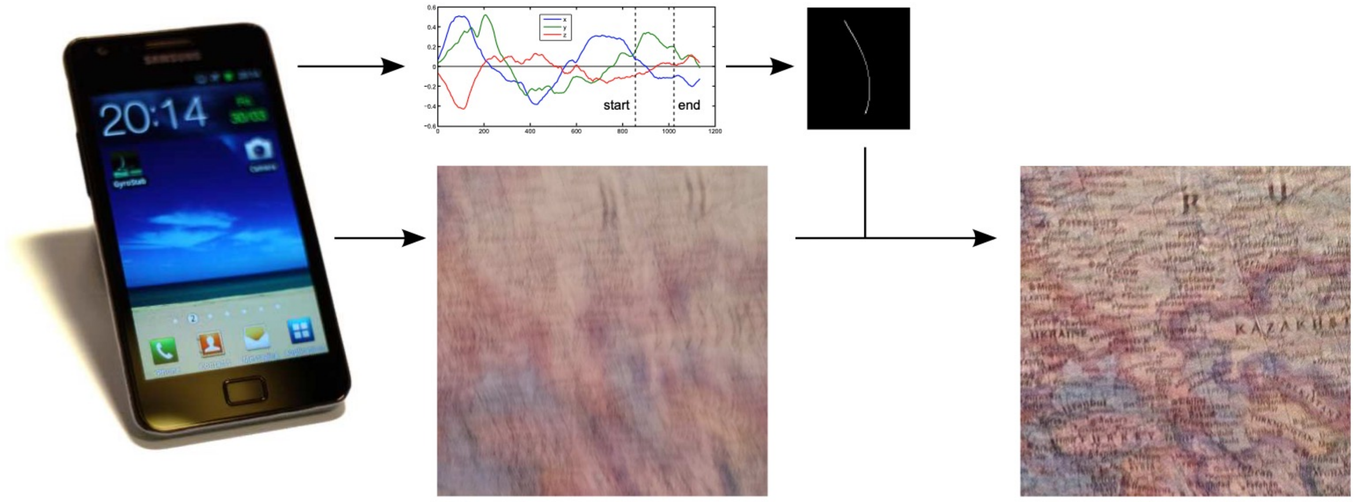



Figure 1: Basic application workflow. Together with a taken photograph gyroscope data are recorded, which is a base for blur kernel estimation. A deconvolution is then performed to remove blur from the image.