Motion Blur and Deconvolution

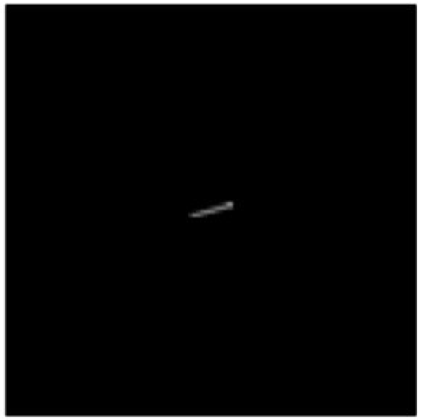
https://liaoux.github.io/uir-cv-2026/

Introduction to Computer Vision
Spring 2026, Lecture 3

Motion Blur

Scene $f(x, y)$

$*$



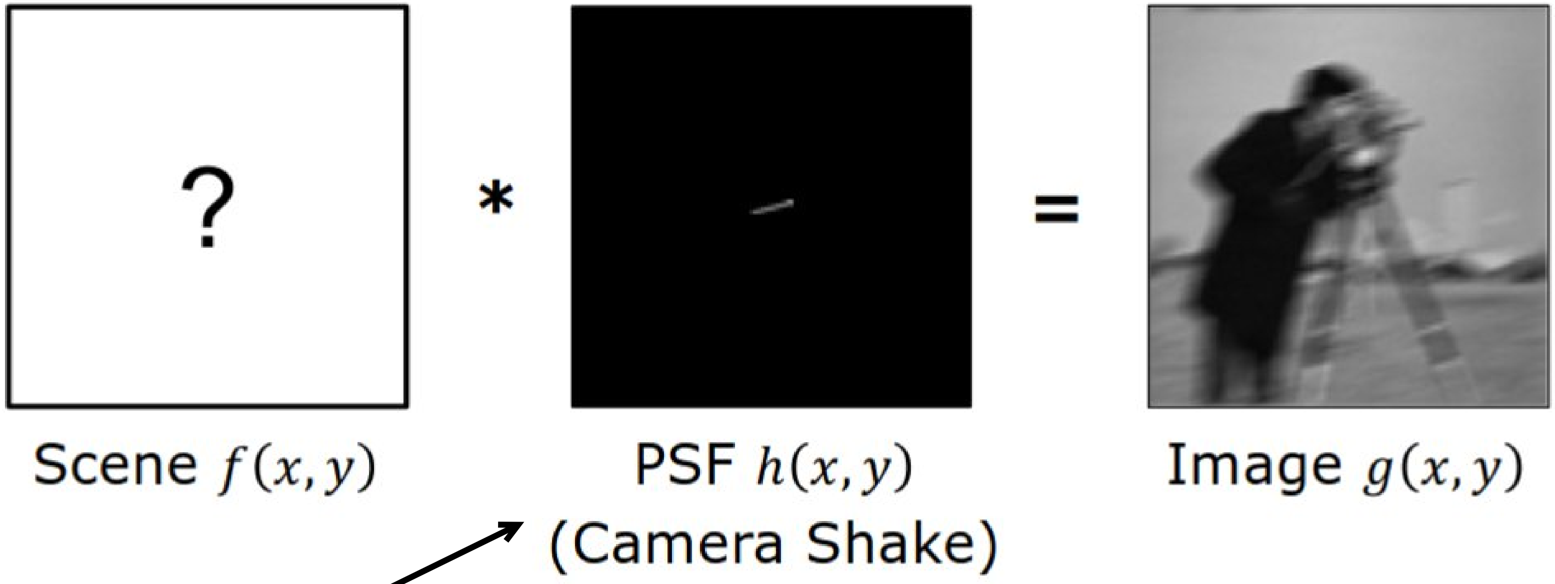
PSF $h(x, y)$
(Camera Shake)

$=$

Image $g(x, y)$

$$f(x, y) * h(x, y) = g(x, y)$$

Motion Blur

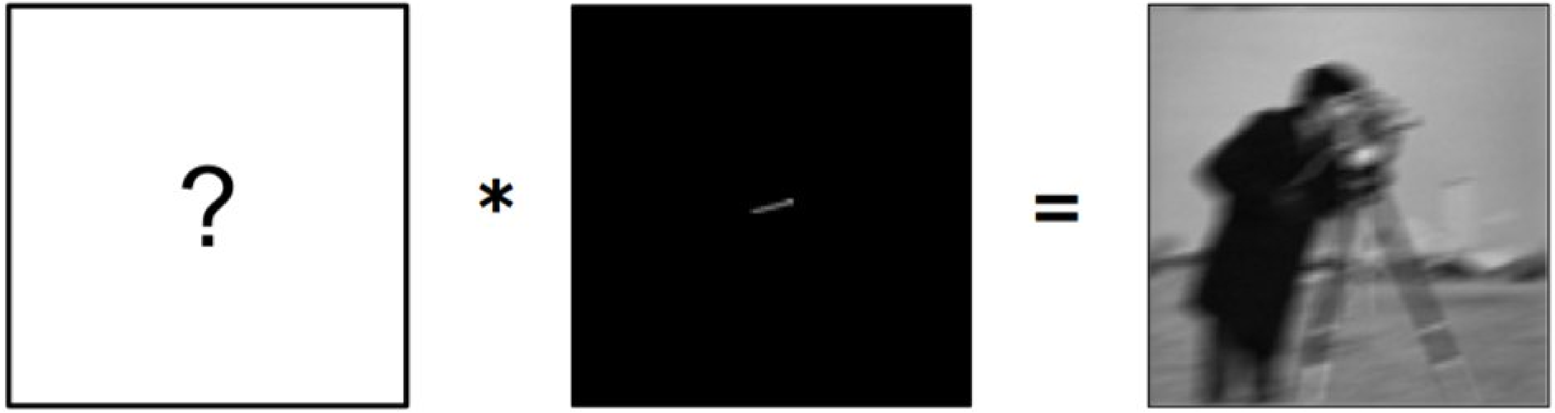


Where this function comes from?

$$f(x,y) * h(x,y) = g(x,y)$$

Given captured image $g(x,y)$ and PSF $h(x,y)$,
can we estimate actual scene $f(x,y)$?

Motion Blur



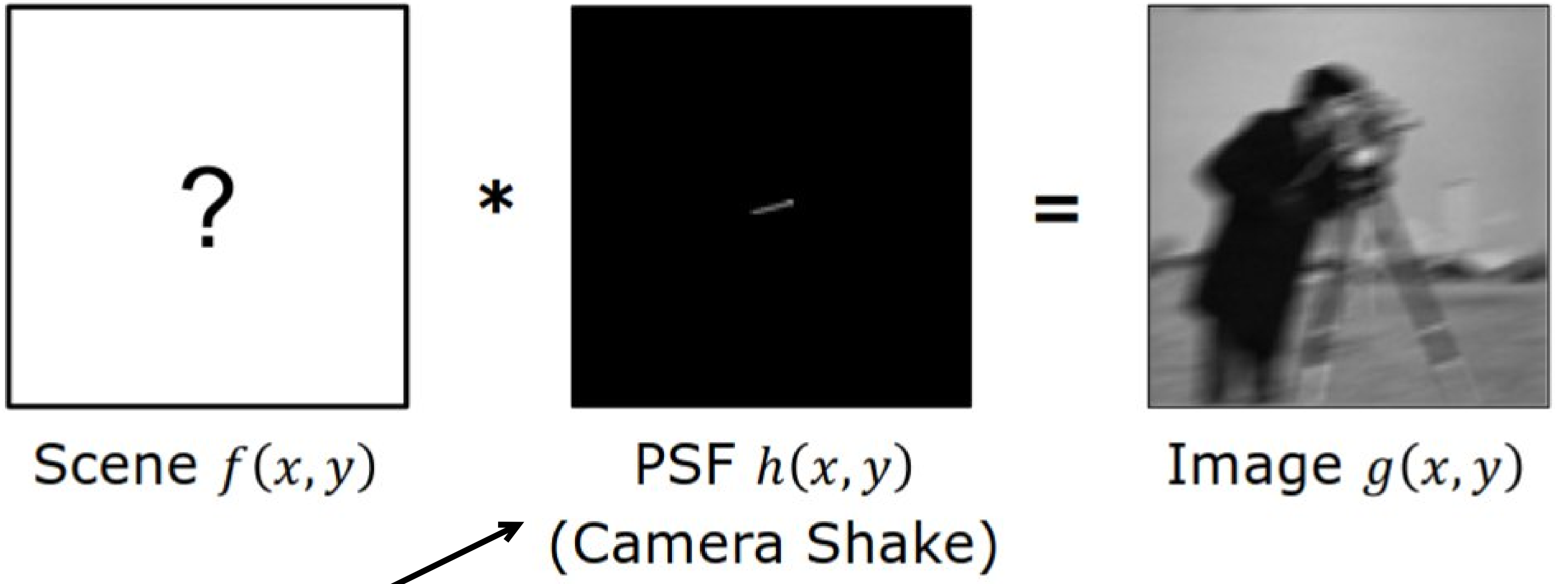
Scene $f(x,y)$ | PSF $h(x,y)$ (Camera Shake) | Image $g(x,y)$

The Point Spread Function h(x,y) can be estimated using an inertial measurement unit, or an IMU, embedded within the capture device (phone, tablet, or camera).

$$f(x,y) * h(x,y) = g(x,y)$$

Given captured image $g(x,y)$ and PSF $h(x,y)$, can we estimate actual scene $f(x,y)$?

Motion Blur



The Point Spread Function h(x,y) can be estimated using an inertial measurement unit, or an IMU, embedded within the capture device (phone, tablet, or camera).

$$f(x,y) * h(x,y) = g(x,y)$$

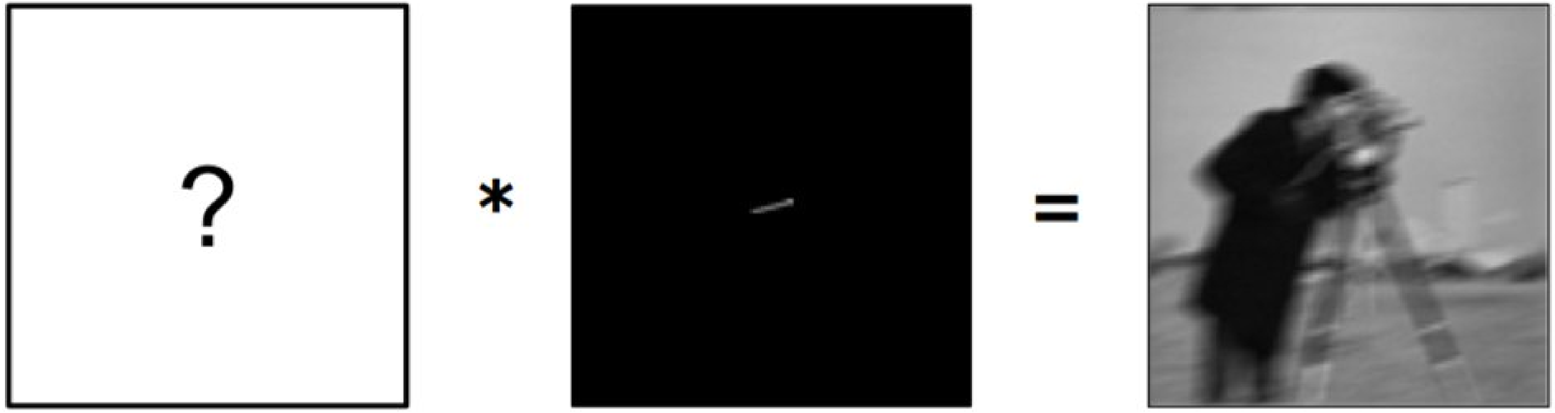
Given captured image $g(x,y)$ and PSF $h(x,y)$, can we estimate actual scene $f(x,y)$?

Fourier Transform to the rescue

Recall: The Convolution Theorem

Spatial Domain		Frequency Domain
$g(x) = f(x) * h(x)$ Convolution	$\longleftrightarrow$	$G(u) = F(u)\, H(u)$ Multiplication
$g(x) = f(x)\, h(x)$ Multiplication	$\longleftrightarrow$	$G(u) = F(u) * H(u)$ Convolution

Motion Deblur: Deconvolution



Scene $f(x,y)$ PSF $h(x,y)$ (Camera Shake) Image $g(x,y)$

Let f' be the recovered scene.

$$f'(x,y) * h(x,y) = g(x,y)$$

$$F'(u,v)H(u,v) = G(u,v)$$

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \longrightarrow \boxed{\text{IFT}} \longrightarrow f'(x,y)$$

Motion Deblur: Deconvolution

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \longrightarrow \boxed{\text{IFT}} \longrightarrow f'(x,y)$$

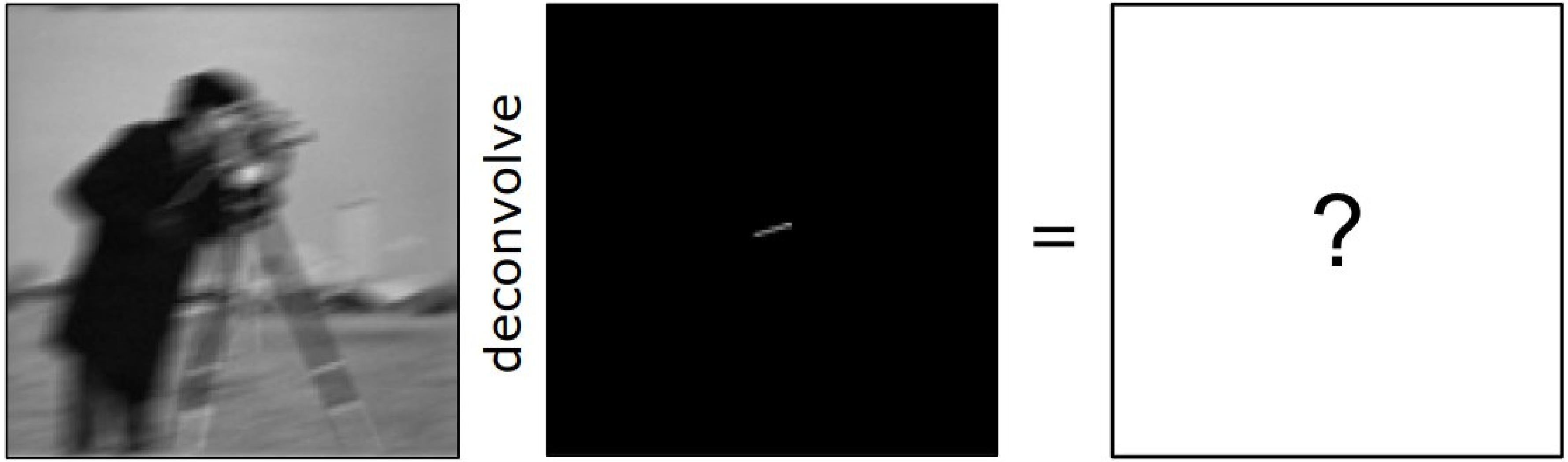
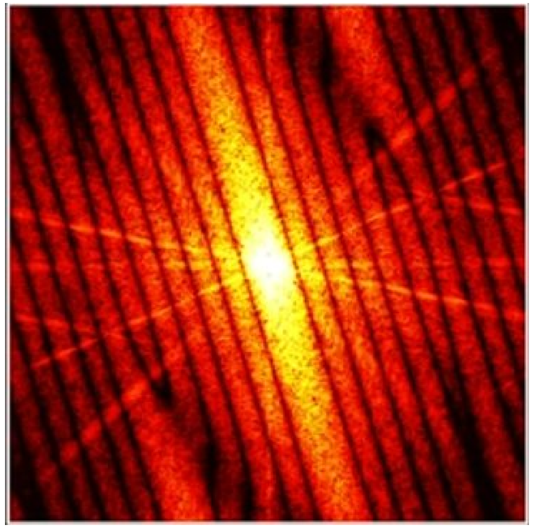


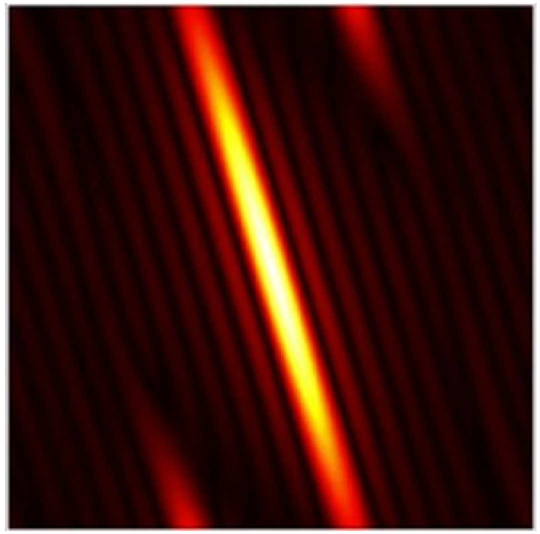
Image $g(x,y)$ PSF $h(x,y)$ Recovered $f'(x,y)$

Motion Deblur: Deconvolution

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \longrightarrow \boxed{\text{IFT}} \longrightarrow f'(x,y)$$



/



=

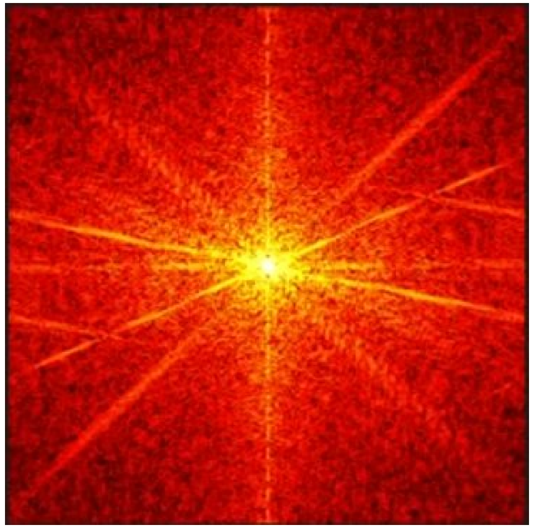


Image $G(u,v)$ | PSF $H(u,v)$ | Recovered $F'(u,v)$

Step 1: Recover $F'(u,v)$ in Fourier Domain

Motion Deblur: Deconvolution

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \longrightarrow \boxed{\text{IFT}} \longrightarrow f'(x,y)$$

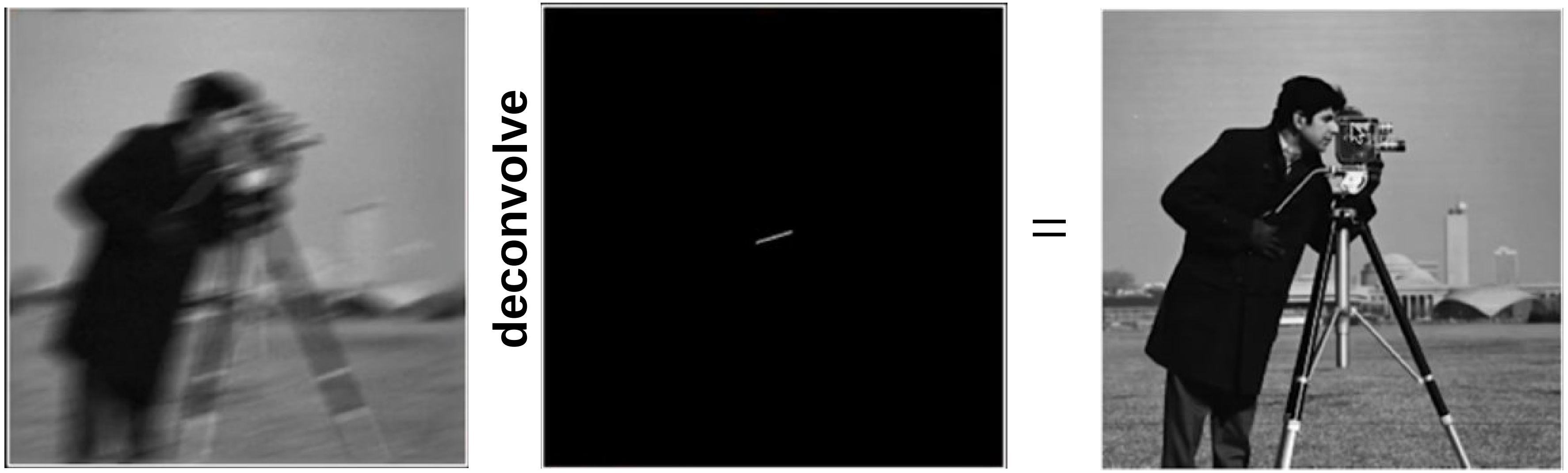
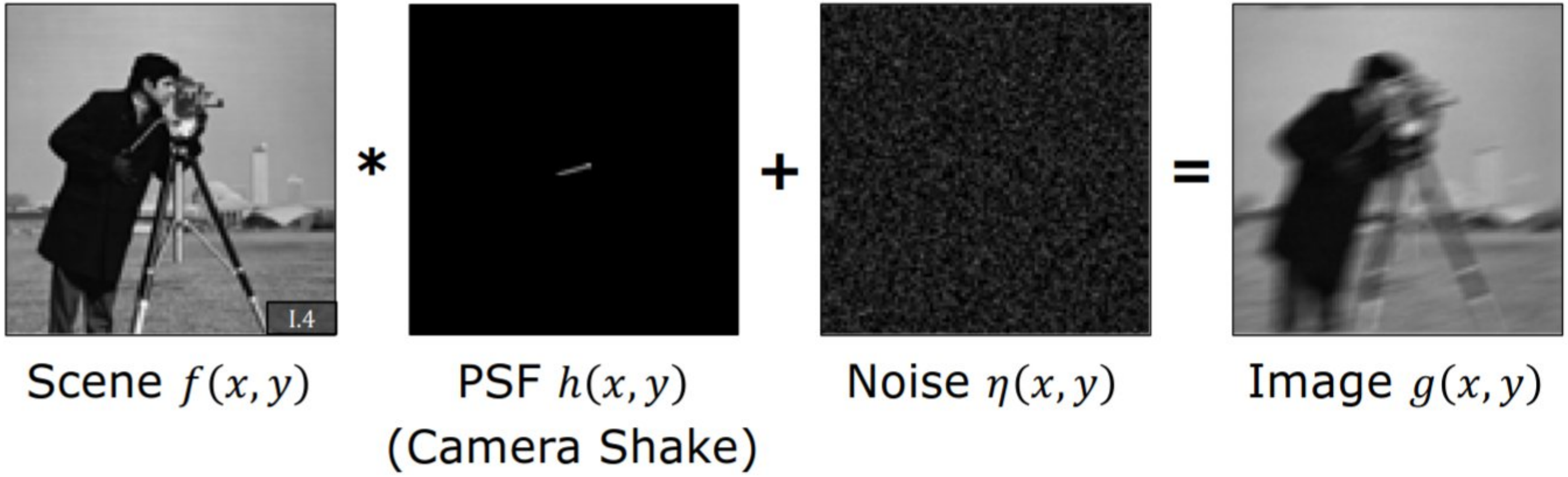


Image $g(x,y)$ PSF $h(x,y)$ Recovered $f'(x,y)$

Step 2: Compute IFT of $F'(u,v)$ to recover scene

Adding Noise to the Problem



$$f(x,y) * h(x,y) + \eta(x,y) = g(x,y)$$

Can we afford to ignore noise?

Types of Noise in Image Sensors

Noise: Unwanted modification of signal during capture, conversion, transmission, processing.

- **Photon Shot Noise (Scene Dependent)**
  - Quantum nature of light
  - Random arrival of photons

Types of Noise in Image Sensors

Noise: Unwanted modification of signal during capture, conversion, transmission, processing.

- **Photon Shot Noise (Scene Dependent)**
  - Quantum nature of light
  - Random arrival of photons

- **Readout Noise (Scene Independent)**
  - Electronic Noise: Pre analog-to-digital conversion
  - Quantization Noise: Post analog-to-digital conversion

Types of Noise in Image Sensors

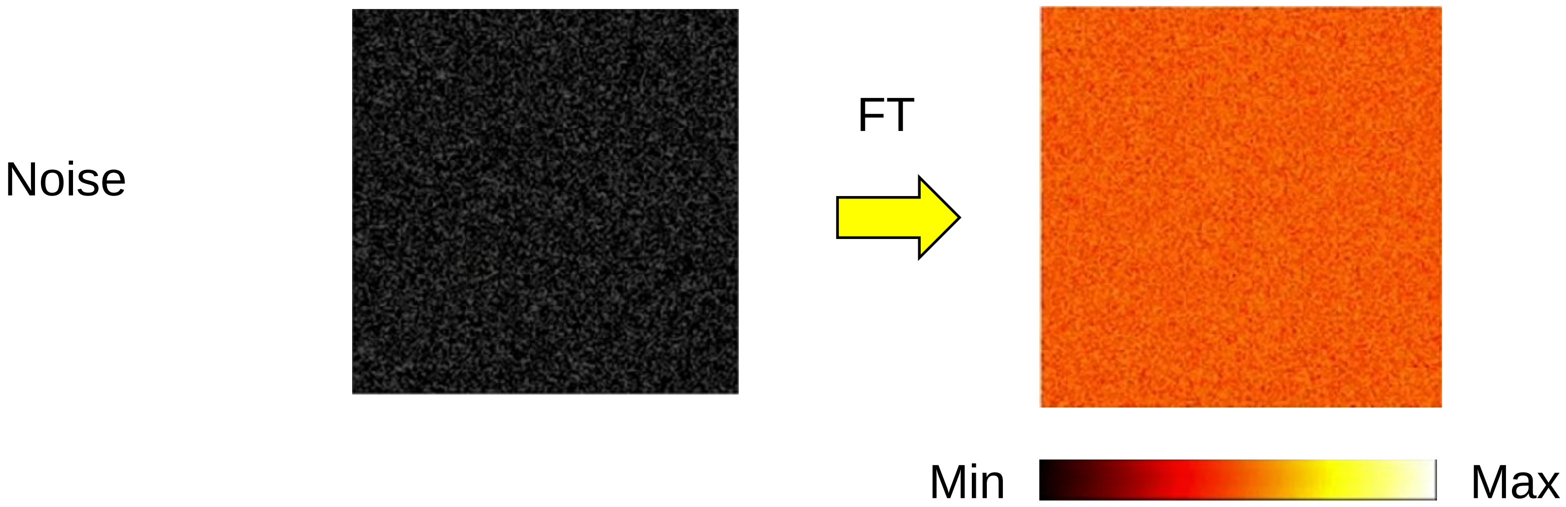
Noise: Unwanted modification of signal during capture, conversion, transmission, processing.

- **Photon Shot Noise (Scene Dependent)**
  - Quantum nature of light
  - Random arrival of photons

- **Readout Noise (Scene Independent)**
  - Electronic Noise: Pre analog-to-digital conversion
  - Quantization Noise: Post analog-to-digital conversion

- **Other Sources (Scene Independent)**
  - Dark Current Noise: Thermally generated electrons
  - Fixed Pattern Noise: Defective pixels

Recall: FT of Noise

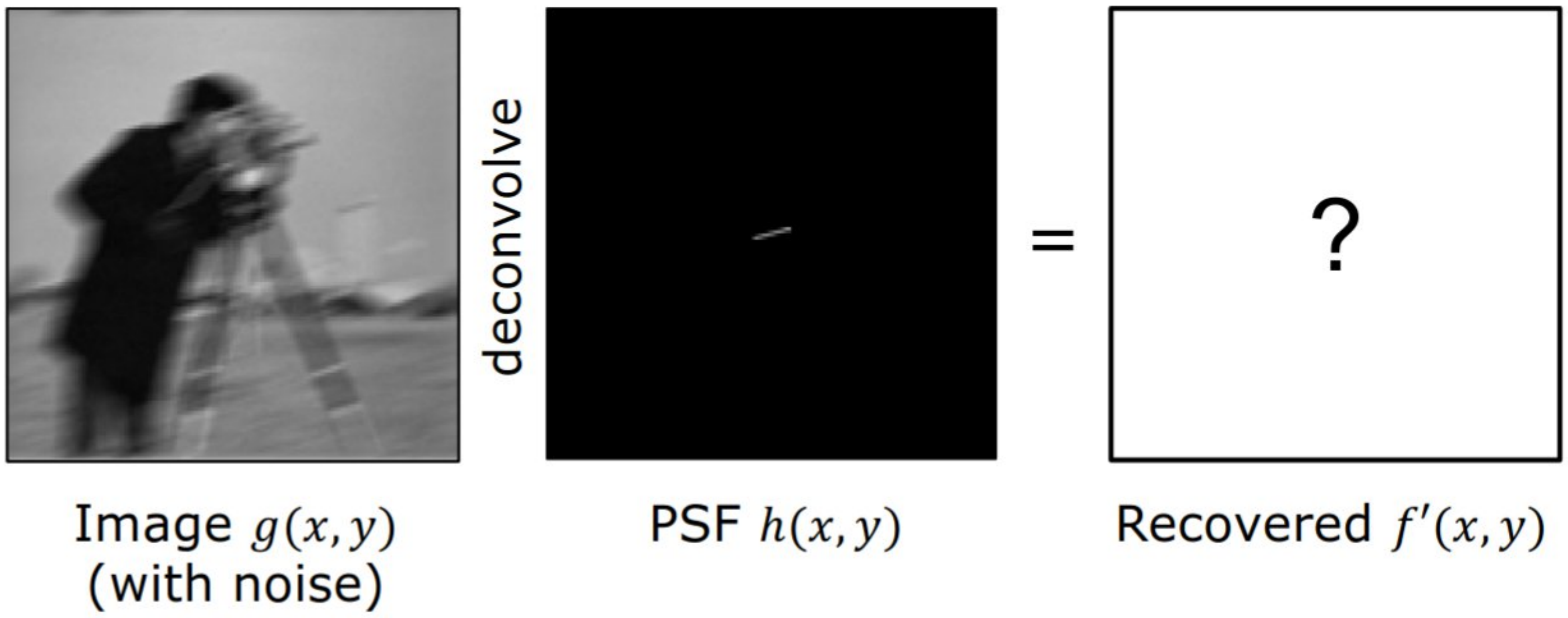


The Fourier Transform of noise produces very strong values in the frequency domain, even for high frequencies

Motion Deblur: Deconvolution

If we ignore the noise $\eta(x,y)$:

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \longrightarrow \boxed{\text{IFT}} \longrightarrow f'(x,y)$$



Motion Deblur: Deconvolution

If we ignore the noise $\eta(x,y)$:

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \longrightarrow \boxed{\text{IFT}} \longrightarrow f'(x,y)$$

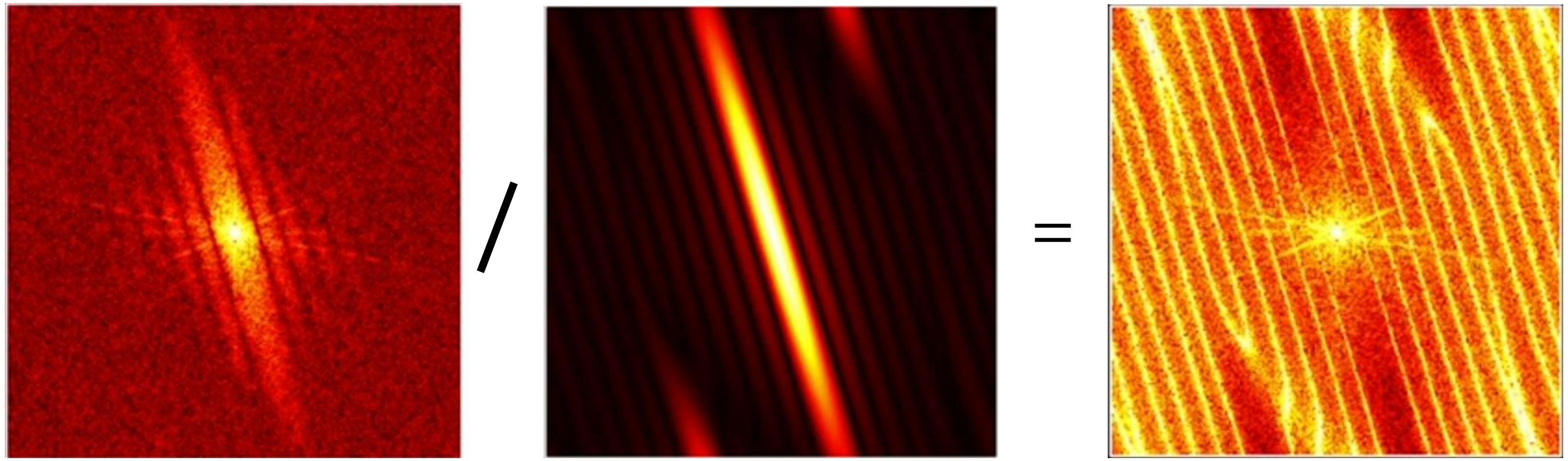


Image $G(u,v)$ PSF $H(u,v)$ Recovered $F'(u,v)$

Higher frequencies in $F'(u,v)$ are amplified

Motion Deblur: Deconvolution

If we ignore the noise $\eta(x,y)$:

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \longrightarrow \boxed{\text{IFT}} \longrightarrow f'(x,y)$$

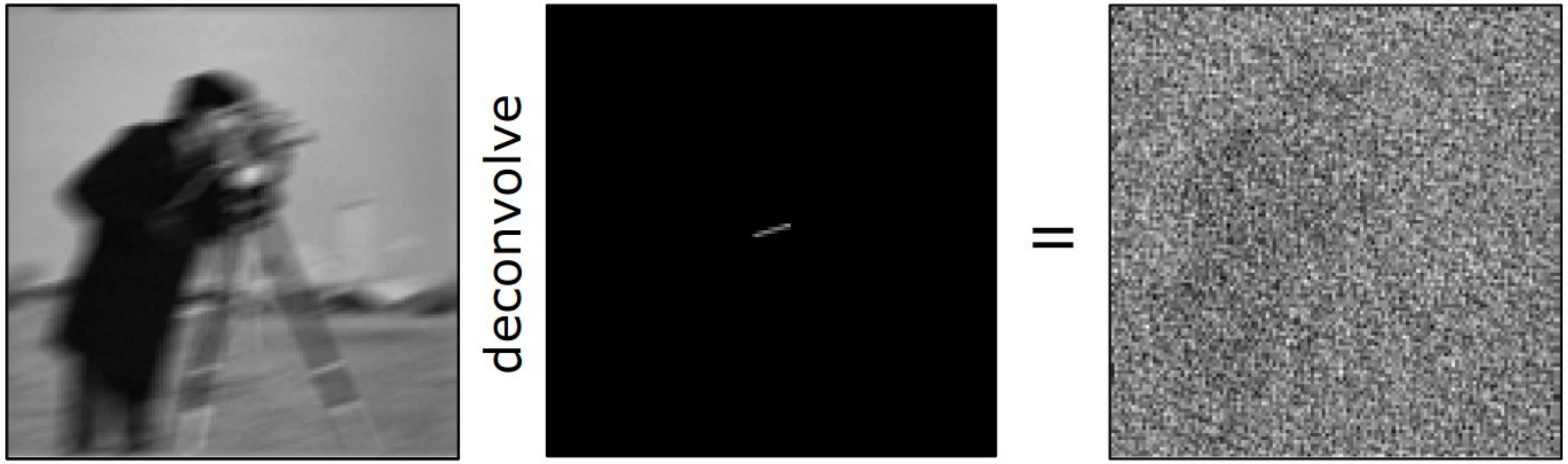


Image $g(x,y)$ (with noise) | PSF $h(x,y)$ | Recovered $f'(x,y)$

Noise is significantly amplified

Deconvolution: Issues

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \longrightarrow \boxed{\text{IFT}} \longrightarrow f'(x,y)$$

1. Where $H(u,v) = 0$, $F'(u,v) = \infty \rightarrow$ Not recoverable

2. Motion blur filter $H(u,v)$ is a low pass filter.

   For high frequencies (u,v):
   - Noise $N(u,v)$ in $G(u,v)$ is high
   - Filter $H(u,v) \approx 0$

   } Noise in $G(u,v)$ is amplified

Deconvolution: Issues

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \longrightarrow \text{IFT} \longrightarrow f'(x,y)$$

1. Where $H(u,v) = 0$, $F'(u,v) = \infty \rightarrow$ Not recoverable

2. Motion blur filter $H(u,v)$ is a low pass filter.

   For high frequencies (u,v):
   - Noise $N(u,v)$ in $G(u,v)$ is high
   - Filter $H(u,v) \approx 0$

   } Noise in $G(u,v)$ is amplified

We need some kind of Noise Suppression.

Noise Suppression: Wiener Deconvolution

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \left[\frac{1}{1 + \frac{NSR(u,v)}{|H(u,v)|^2}} \right]$$

Where:

Noise-to-Signal Ratio, $NSR(u,v)$:

$$NSR(u,v) = \frac{\text{Power of Noise at } (u,v)}{\text{Power of Signal (Scene) at } (u,v)} = \frac{|N(u,v)|^2}{|F(u,v)|^2}$$

Noise Suppression: Wiener Deconvolution

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} = G(u,v) \, . \, \mathbf{W(u,v)}$$

Where:

Weiner Filter $\stackrel{\text{def}}{=}$ $$W(u,v) = \frac{1}{H(u,v)}\left[\frac{1}{1 + \frac{NSR(u,v)}{|H(u,v)|^2}}\right]$$

Noise Suppression: Wiener Deconvolution

$$F'(u,v) = \frac{G(u,v)}{H(u,v)} \left[\frac{1}{1 + \frac{NSR(u,v)}{|H(u,v)|^2}} \right]$$

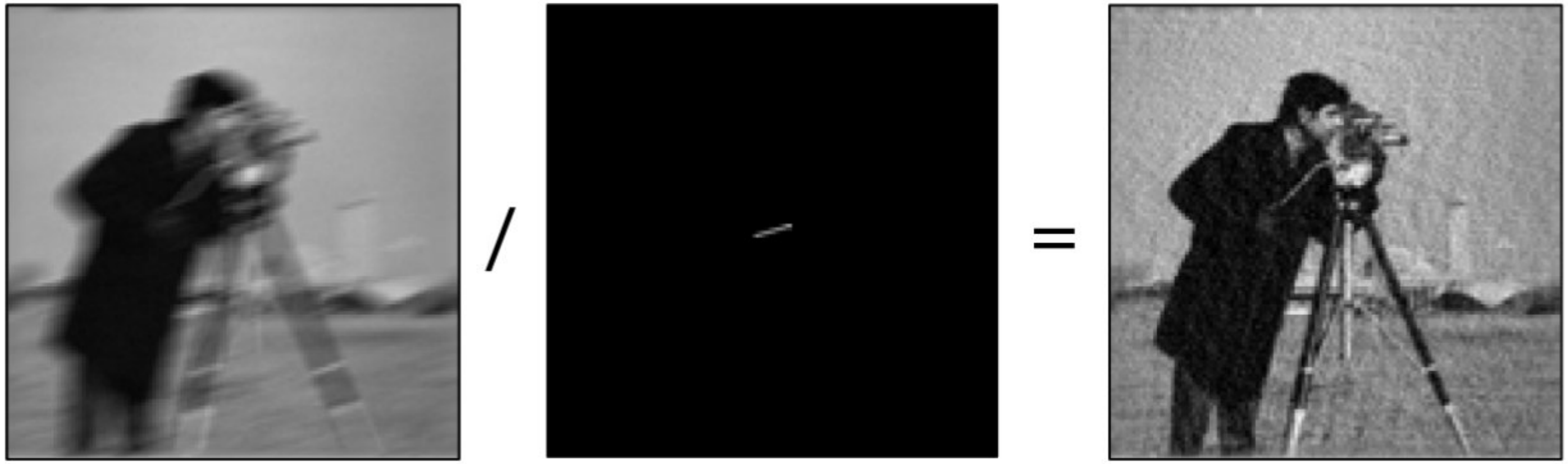
- Determining NSR requires us to have prior knowledge of the noise "pattern" and the scene (or of a similar scene).

$$NSR(u,v) = \frac{|N(u,v)|^2}{|F(u,v)|^2}$$

- Often NSR is set to a single suitable constant λ.

$$NSR(u,v) = \lambda$$

Noise Suppression: Wiener Deconvolution

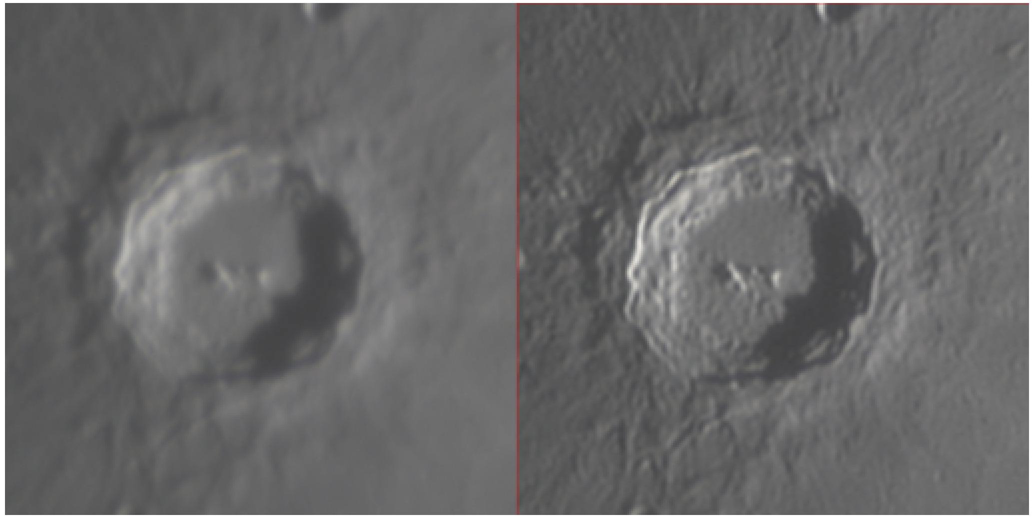


Noisy, Blurred Image $g(x, y)$ PSF $h(x, y)$ Recovered $f'(x, y)$

$NSR(u, v) = \lambda = 0.002$ was used to recover image

Applications of Deconvolution

Astronomy & Telescope Imaging



Atmospheric turbulence and optical imperfections cause blur in telescope images. Deconvolution is used to sharpen images of planets, galaxies, and lunar surfaces — as illustrated by the Copernicus crater example below.

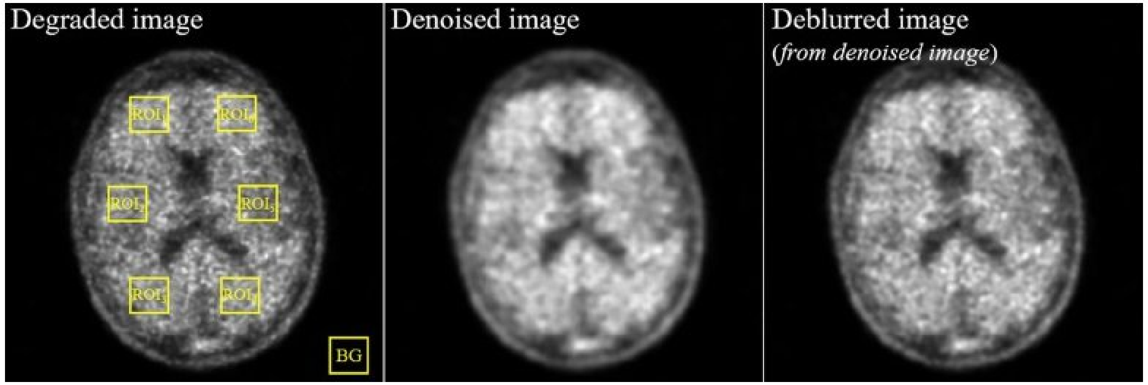
Applications of Deconvolution

Vehicle License Plate Motion Deblurring

Motion blur is a convolution applied to the sharp image by the camera during exposure. Deconvolution mathematically inverts this process to recover the original sharp plate.

Applications of Deconvolution

Medical Imaging



Deconvolution is applied in CT, MRI, and PET scans to improve spatial resolution and remove system-induced blur. In PET imaging for example, the scanner's PSF smears out radioactive tracer distributions, and deconvolution recovers sharper organ boundaries.