The University of Texas at Austin
Electrical and Computer Engineering
Cockrell School of Engineering

Spring 2022

INTRODUCTION TO COMPUTER VISION

Atlas Wang
Assistant Professor, The University of Texas at Austin

Visual Informatics Group@UT Austin
https://vita-group.github.io/

Visual Degradation

Data Acquisition

Degradation before Data Acquisition

- Heavy Rain/Snow
- Underwater
- Low Light
- Haze/Sandstorm

Degradation in Data Acquisition

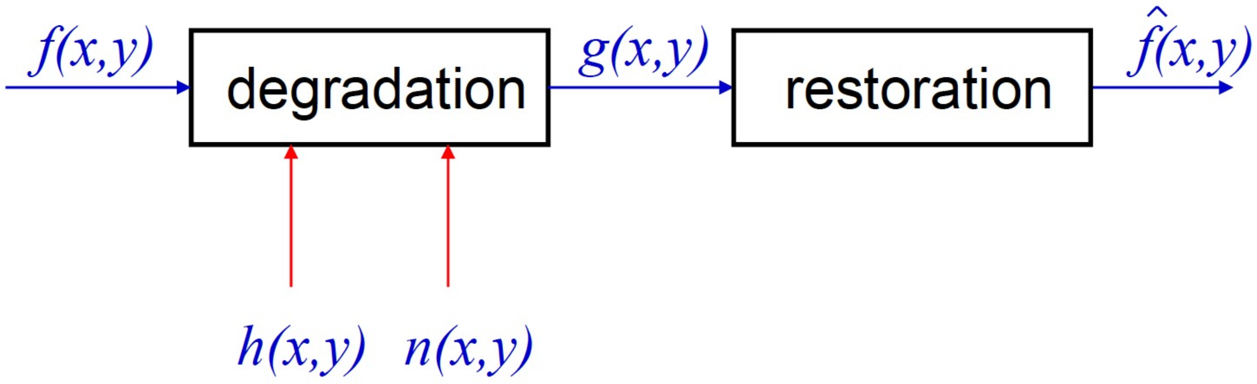
- Downsample
- Motion Blur
- System Noise
- Optical Distortion

Degradation after Data Acquisition

- Scratches
- Watermark
- Mildew
- Compression Loss

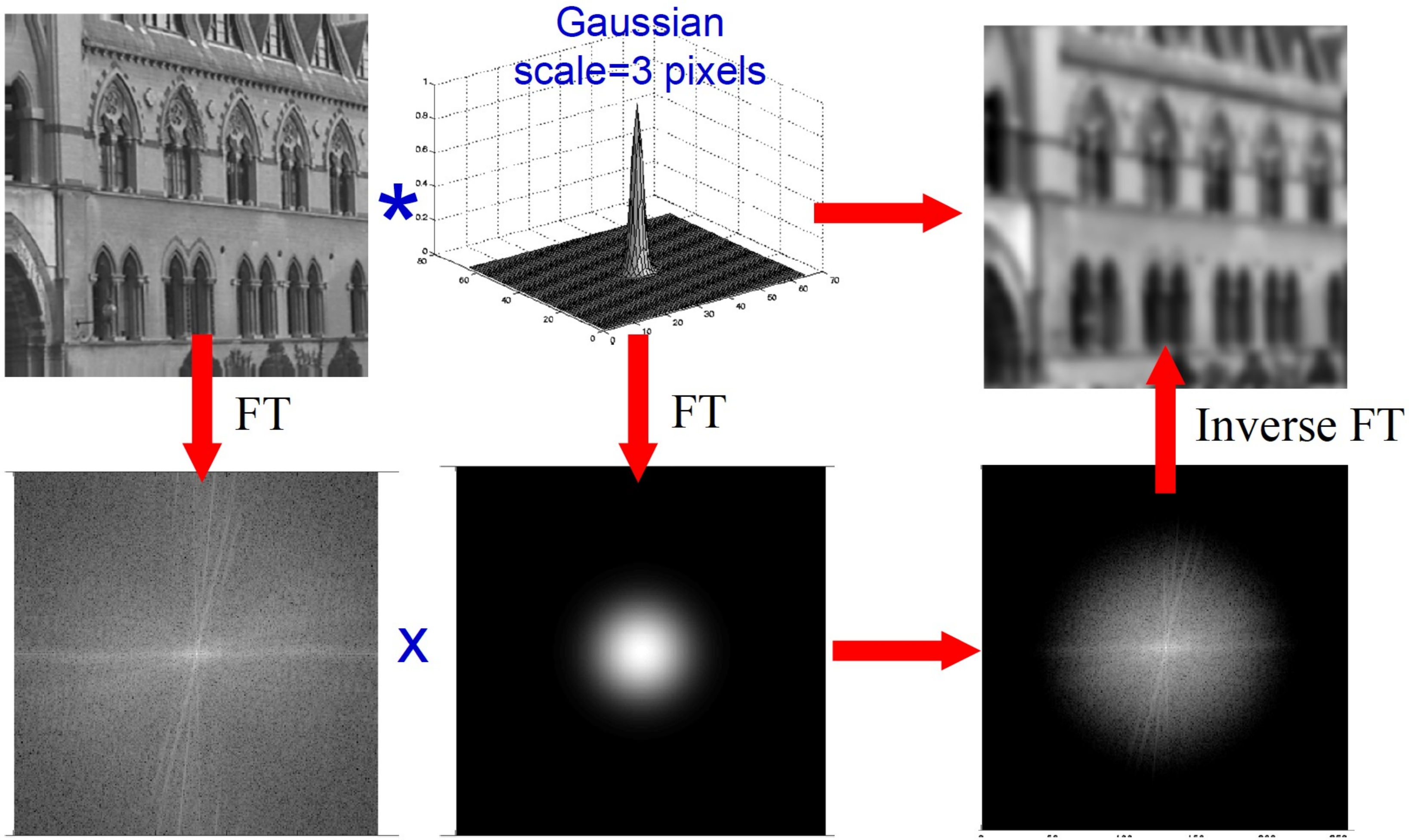
Image Degradation Model

- $f(x,y)$ – image before degradation, 'true image'
- $g(x,y)$ – image after degradation, 'observed image'
- $h(x,y)$ – degradation filter
- $\hat{f}(x,y)$ – estimate of $f(x,y)$ computed from $g(x,y)$
- $n(x,y)$ – additive noise



$$g(x,y) = h(x,y) * f(x,y) + n(x,y) \Leftrightarrow \quad G(u,v) = H(u,v)\, F(u,v) + N(u,v)$$

Example: Image Blur



Blurring acts as a low pass filter and attenuates higher spatial frequencies

Goal of Image Enhancement Diversified

- From traditional signal processing (reconstruction) viewpoint
  - Full-reference metrics: PSNR, SSIM, etc.

- … to human perception (subjective quality)-based
  - No-reference metrics (e.g., NIQE), and human study

- … And to task-oriented, “end utility”-based
  - Typical examples: dehazing, deraining, (extreme) light, underwater …
  - **Representative datasets: RESIDE** dehazing (TIP’18), **MPID** deraining (CVPR’19)
  - **CVPR UG2+ Challenge:** http://www.ug2challenge.org

Discussion: Patch-Based v.s. Image-Level

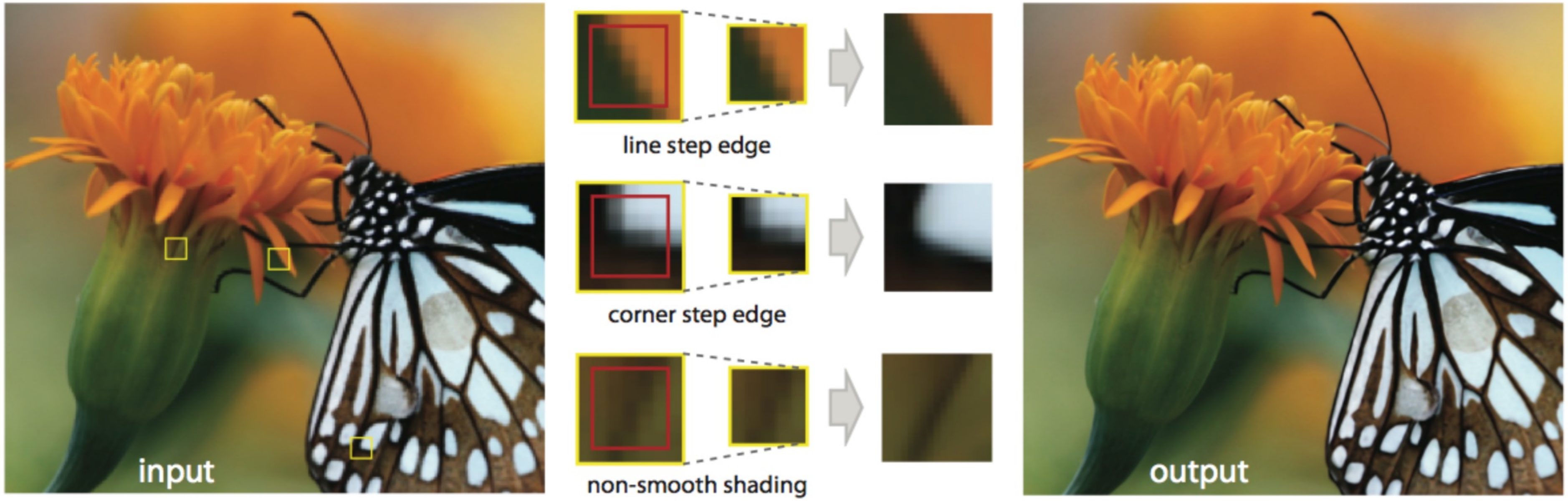
- The term “**patch**-**based**” may be vague because it can refer to any algorithm that works with small **image patches**.
  - BM3D image denoising, sparse coding for image super-resolution, image compression algorithms such as JPEG…

- Traditional image processing works on patches
  - Efficiency (esp. when model learning capacity is limited)
  - A lot of natural image statistics and similarities to exploit

- Deep learning image processing works on whole images
  - Mostly obtain better results as they are more “global-view”
  - But often ignore some useful prior knowledge on patch-level

Discussion: Self v.s. External Similarity

- Natural images contain abundant **self-similarities**.
  - For every patch in a natural image, we can probably find many similar patches in the same image.
  - Nonlocal patch-based methods exploit this self-similarity by finding/collecting similar patches and processing them jointly.
  - Cross-scale self-similarity (*Example Below*)



Learning to Enhance Images

- Data-driven training of “end-to-end” models (usually assuming “pairs”)
- Prior/physical information can still be helpful

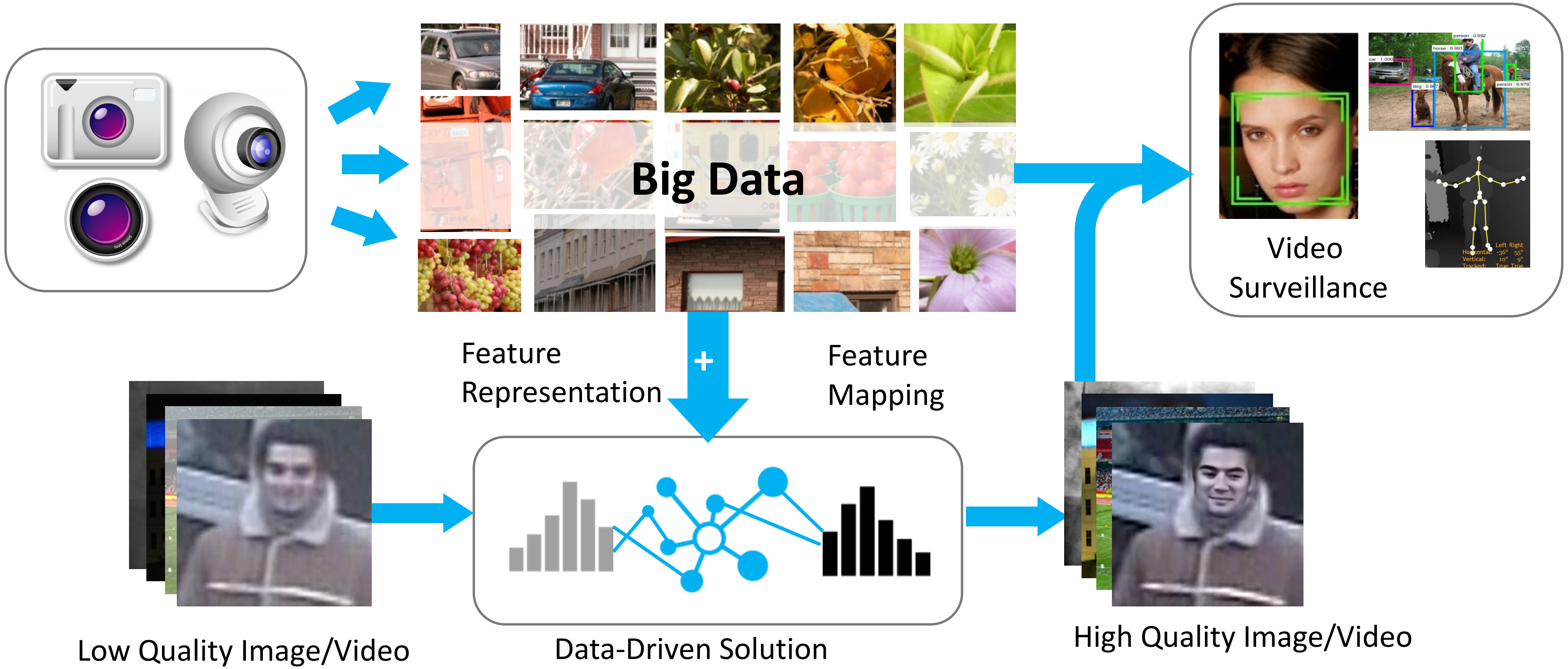


Image Denoising

- Simplest Low-Level Vision Problem

  - Noisy Measurement: $y = x + e$

=

+

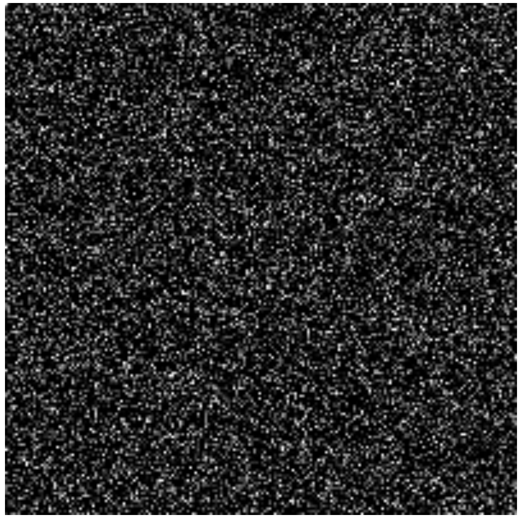


Image Denoising

- Simplest Low-Level Vision Problem

  - Estimate the clean image: $\hat{\boldsymbol{x}} = f(y)$

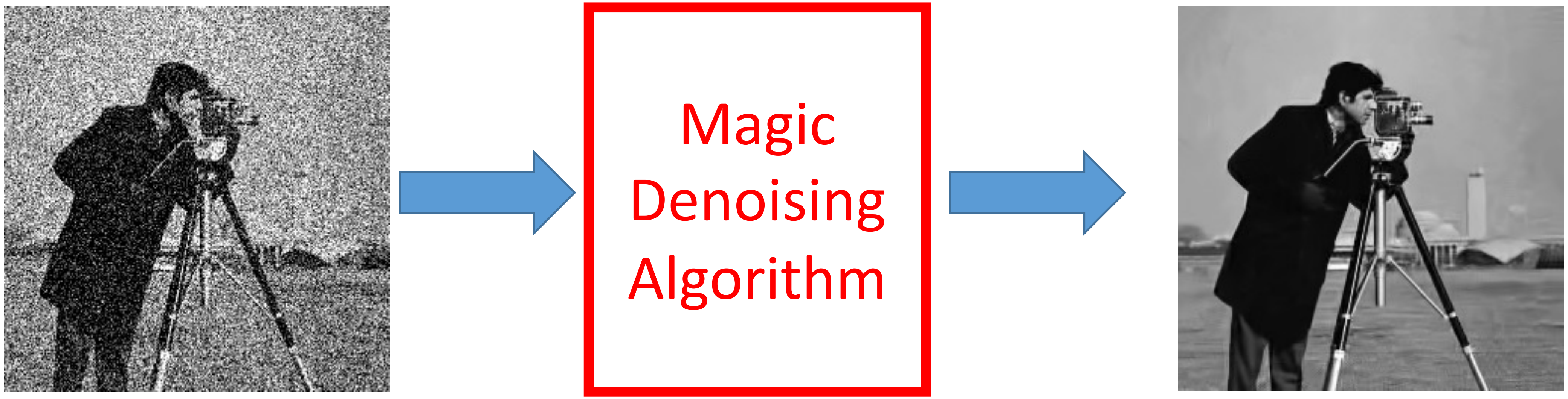


Image Denoising – Conventional Methods

- Collaborative Filtering
  - Non-local Mean, BM3D, etc

Classical Image Denoising: BM3D

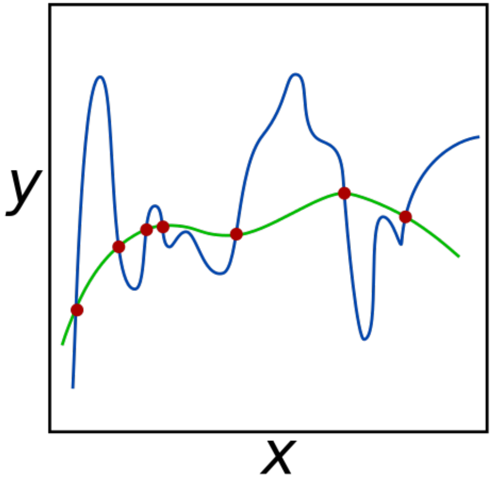
- BM3D = *Block-Matching and 3D filtering*, suggested first in 2007.
- Given a 2D square-block, finds all 2D similar blocks and “group” them together as a 3D array, then performs a *collaborative filtering* (method that the authors designed) of the group to obtain a noise-free 2D estimation.
- Averaging overlapping pixels estimations.
- Gives state of the art results.

Based on: K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. *Image denoising by sparse 3-D transform-domain collaborative filtering*. IEEE Transactions on Image Processing, 16(8):2080–2095, 2007.

Patch-based + Self-Similarity + Domain Expertise

Image Denoising – Conventional Methods

- Collaborative Filtering
  - Non-local Mean, BM3D, etc
- Piece-wise Smooth
  - Total Variation, Tikhonov Regularization, etc



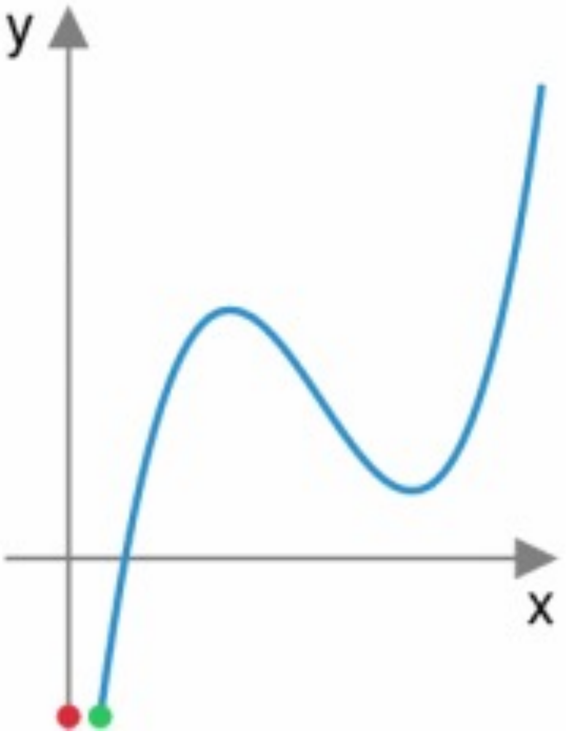
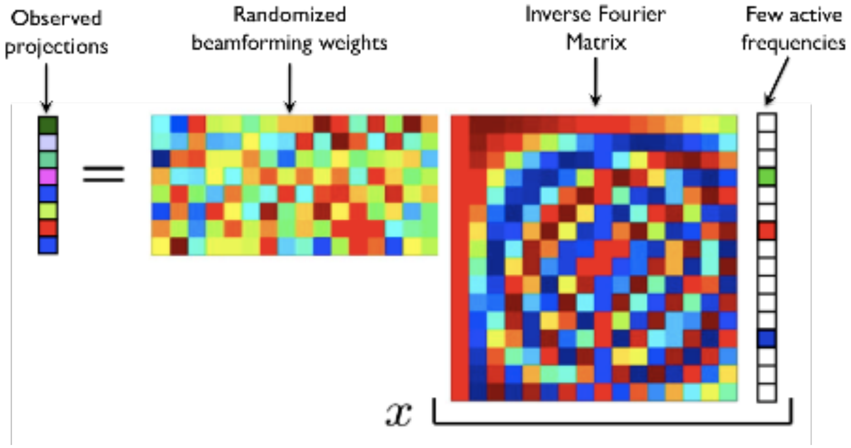
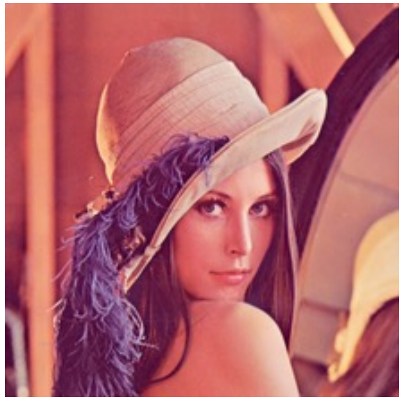
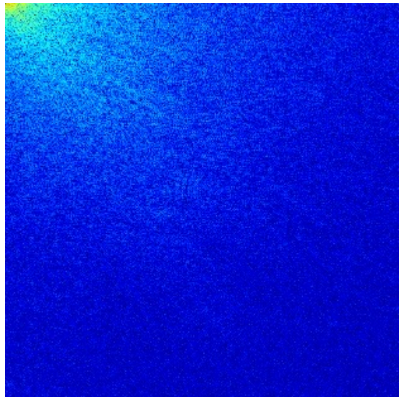


Image Denoising – Conventional Methods

- Collaborative Filtering
  - Non-local Mean, BM3D, etc
- Piece-wise Smooth
  - Total Variation, Tikhonov Regularization, etc
- Sparsity
  - Discrete Cosine Transform (DCT), Wavelets, etc
  - Dictionary Learning: KSVD, OMP, Lasso, etc
  - Analysis KSVD, Transform Learning, etc







It is all about good "prior"

Conventional

- Shallow Model
  - Equivalently one free layer

Deep Learning

- Deep Model
  - Multiple free layers

Conventional

- Shallow Model
  - One free layer
- **Unsupervised**
  - **No training corpus needed**
  - **Data efficient**

Deep Learning

- Deep Model
  - Multiple free layers

- **Supervised**
  - **Training corpus needed**
  - **Data inefficient**

?

Conventional

- Shallow Model
  - One free layer
- Unsupervised
  - No training corpus needed
  - Data efficient
- **Inverse Problem**
  - **Assumption & Understanding of the Data**
  - **Regularizer & structures of the Model**
  - **Flexible**

Deep Learning

- Deep Model
  - Multiple free layers
- Supervised
  - Training corpus needed
  - Data inefficient
- **Inverse Problem**
  - **Little assumption**
  - **Almost free model**
  - **Few work until recent**

Image Deblurring

- Blurred Measurement: $y = M \otimes \boldsymbol{x}$

=

$\otimes$

$$M = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Image Deblurring

- Estimate the stable image: $\hat{\boldsymbol{x}} = f(y)$

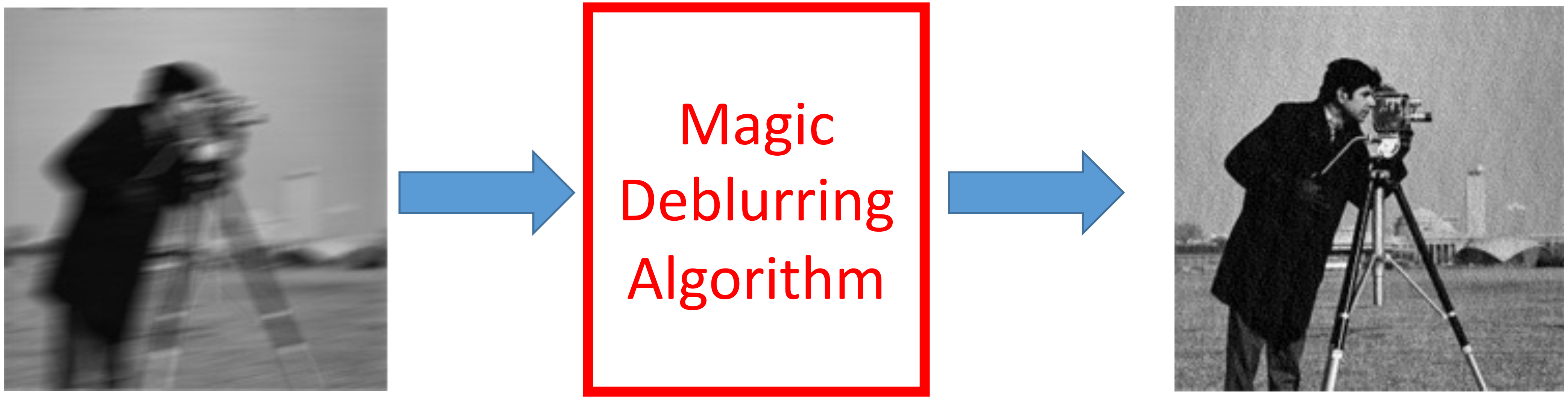


Image Deblurring

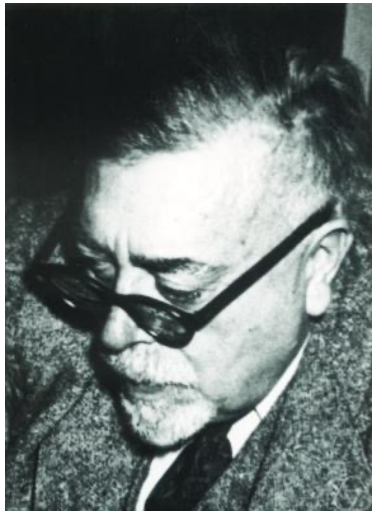
- Non-blind Image Deblurring
  - Suppose you know the blurring kernel, $\boldsymbol{M}$.
  - $\widehat{\boldsymbol{x}} = f(y, M)$
  - All training data need to have consistent $\boldsymbol{M}$, as the testing data

Image Deblurring

- Non-blind Image Deblurring
  - Suppose you know the blurring kernel, $\boldsymbol{M}$.
  - $\hat{\boldsymbol{x}} = f(y, M)$
  - All training data need to have consistent $\boldsymbol{M}$, as the testing data
- Blind Image Deblurring – More challenging yet practical problem
  - Estimate both the image, and the blurring kernel
  - $\{\hat{\boldsymbol{x}}, M\} = f(y)$

Wiener Filtering

Norbert Wiener
(1894-1964)
"Father of cybernetics"



Restoration with a Wiener filter

$$G(u,v) = H(u,v)\,F(u,v) + N(u,v)$$

$$\hat{F}(u,v) = W(u,v)\,G(u,v)$$

$$\hat{F}(u,v) = W(u,v)\,G(u,v)$$

$$W(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 + K(u,v)}$$

where

$$K(u,v) = S_\eta(u,v)/S_f(u,v)$$

$S_f(u,v) = |F(u,v)|^2$ power spectral density of $f(x,y)$

$S_\eta(u,v) = |N(u,v)|^2$ power spectral density of $\eta(x,y)$

Limitation: Assuming known stationary signal and noise spectra, and additive noise

$$g(x,y) \rightarrow \boxed{\text{F.T.}} \xrightarrow{G(u,v)} \boxed{\text{Wiener filter}} \xrightarrow{\hat{F}(u,v)} \boxed{\text{I.F.T.}} \xrightarrow{\hat{f}(x,y)}$$

Example: Motion Deblurring by Wiener Filtering

blur = 20 pixels

$$W(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 + K(u,v)}$$

1. Compute the FT of the blurred image
2. Multiply the FT by the Wiener filter
3. Compute the inverse FT

$$\hat{F}(u,v) = W(u,v)\, G(u,v)$$

Maximum a posteriori (MAP) Estimation

- 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.

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

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

Example

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

to suppress high frequency noise

Blind Deblurring?

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

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

Blind Deblurring

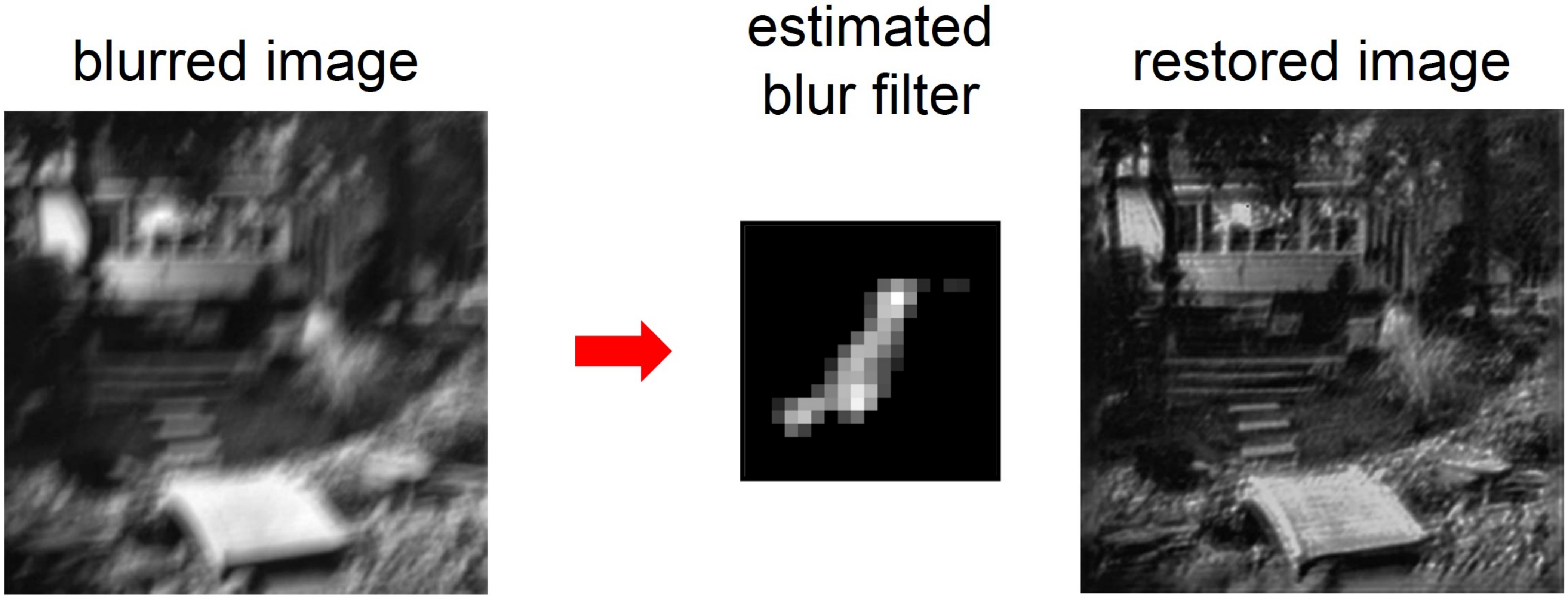


Image Super-Resolution

- Low-Resolution Measurement: $y = D * M \otimes \boldsymbol{x}$

=

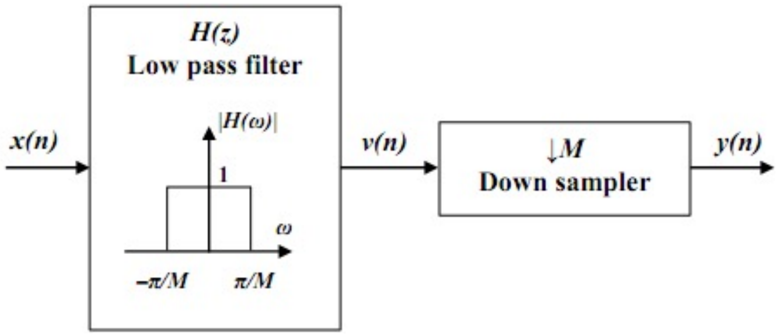


Image Super-Resolution

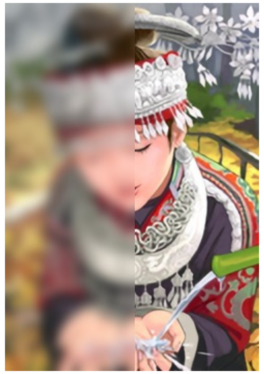
- Estimate the stable image: $\hat{\boldsymbol{x}} = f(y)$

Many More Tasks in the Real World!

Underwater Enhancement

Dehazing

Inpainting



Super Resolution

Rain Removal

Denoising

Low Light Enhancement

• • • • • • • •

The University of Texas at Austin
Electrical and Computer Engineering
Cockrell School of Engineering