Noise, Image Reconstruction with Noise

EE367/CS448I: Computational Imaging and Display
stanford.edu/class/ee367
Lecture 10

Gordon Wetzstein
Stanford University
Topics

- Fixed pattern noise
- Gaussian noise
  - Image reconstruction using MAP
- Poisson noise
  - Richardson-Lucy algorithm
  - RL + TV prior
- The SNR with three types of noise sources
What’s a Pixel?

photon to electron converter
→ photoelectric effect!
What’s a Pixel?

- microlens: focus light on photodiode
- color filter: select color channel
- quantum efficiency: ~50%
- fill factor: fraction of surface area used for light gathering
- photon-to-charge conversion and analog-to-digital conversion (ADC) include noise!

source: Molecular Expressions
ISO ("film speed")

Sensitivity of sensor to light – digital gain

bobatkins.com
Noise

- Noise is (usually) bad!

- Many sources of noise: heat, electronics, amplifier gain, photon to electron conversion, pixel defects, read, …

- Let’s start with something simple: fixed pattern noise
Fixed Pattern Noise

- Dead or “hot” pixels, dust, pixel sensitivity variations ...
- remove with dark frame calibration:

\[ I = \frac{I_{\text{captured}} - I_{\text{dark}}}{I_{\text{white}} - I_{\text{dark}}} \]

on RAW image! not JPEG (nonlinear)
Noise

• Other than that, different noise follows different statistical distributions, these two are crucial:
  • Gaussian
  • Poisson
Gaussian Noise

- Thermal, read, amplifier
- Additive, signal-independent, zero-mean
Gaussian Noise

With i.i.d. Gaussian noise:

\[ b = x + h \sim N(0, \sigma^2) \]

We want to find \( x \)
Gaussian Noise

- With i.i.d. Gaussian noise:

  \[ b = x + h \sim N(0, \sigma^2) \]

\[ x \sim N(x, 0) \]

\[ \sim N(0, 2) \]

\[ b \sim N(x, 2) \]

\[ p(b | x, 2) = \frac{1}{\sqrt{2^2}} e^{-\frac{(b-x)^2}{2^2}} \]
Gaussian Noise

- With i.i.d. Gaussian noise:

\[ b = x + h \sim N(0, s^2) \]

\[ x \sim N(x, 0) \]

- Bayes' rule:

\[ p(x|b, \sigma) \propto p(b|x, \sigma) p(x) \]

\[ b \sim N(x, \sigma^2) \]

\[ p(b|x, \sigma) = \frac{1}{\sqrt{2\pi \sigma^2}} e^{-\frac{(b-x)^2}{2\sigma^2}} \]
Gaussian Noise - MAP

- With i.i.d. Gaussian noise:

\[ b \sim N(x, \sigma^2) \]

\[ p(b|x, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(b-x)^2}{2\sigma^2}} \]

- Bayes’ rule:

\[ p(x|b, \sigma) \propto p(b|x, \sigma) p(x) \]

- Maximum-a-posteriori estimation:

\[ x_{\text{MAP}} = \text{arg max}_{\{x\}} \log (p(x|b, \sigma)) = \text{arg max}_{\{x\}} \log (p(b|x, \sigma) p(x)) \]

\[ = \text{arg min}_{\{x\}} -\log (p(b|x, \sigma)) - \log (p(x)) \]

\[ = \text{arg min}_{\{x\}} \frac{1}{2\sigma^2} \|b - x\|^2 + \log (p(x)) \]

\[ = \text{arg min}_{\{x\}} \frac{1}{2\sigma^2} \|b - x\|^2 + \Gamma(x) \]

\[ x \sim N(x, 0) \]

\[ b = x + \]

\[ \sim N(0, \sigma^2) \]

Some information, a prior, for the image
Gaussian Noise – MAP “Flat” Prior

- Trivial solution (not useful in practice):

$$x_{flat} = \arg \min_{\{x\}} \frac{1}{2\sigma^2} \| b - x \|_2^2 = b$$

$$p(x) = 1$$
Gaussian Noise – MAP Self Similarity Prior

- Gaussian “denoisers” like non-local means and other self-similarity priors actually solve this problem:

\[ x_{NLM} = \arg \min_{\{x\}} \frac{1}{2\sigma^2} \|b - x\|_2^2 + \Gamma(x) = \text{NLM}(b, \sigma^2) \]
General Self Similarity Prior

• Generic proximal operator for function $f(x)$:

$$\text{prox}_{f, \rho} (v) = \arg \min_{x} f(x) + \frac{\rho}{2} \|v - x\|_2^2$$

• Proximal operator for some image prior $\lambda \Gamma(x)$:

$$\text{prox}_{\Gamma, \rho} (v) = \arg \min_{x} \lambda \Gamma(x) + \frac{\rho}{2} \|v - x\|_2^2$$
General Self Similarity Prior

- We can use self-similarity as general image prior (not just for denoising)

\[
\text{prox}_{\text{NLM}, \rho} (v) = \text{NLM} \left( v, \frac{\lambda}{\rho} \right)
\]

\[
\frac{\lambda}{\rho} = \sqrt{\frac{\lambda}{\rho}}
\]

(h parameter in most NLM implementations is std. dev.)

- Proximal operator for some image prior \( \lambda \Gamma (x) \)

\[
\text{prox}_{\Gamma, \rho} (v) = \arg \min_{\{x\}} \lambda \Gamma (x) + \frac{\rho}{2} \| v - x \|_2^2
\]
Image Reconstruction with Gaussian Noise

- With i.i.d. Gaussian noise:

\[ b \sim N(Ax, \sigma^2) \]

\[ p(b \mid x, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(b - Ax)^2}{2\sigma^2}} \]

\[ Ax \sim N(Ax, 0) \sim N(0, \sigma^2) \]

- Bayes’ rule:

\[ p(x \mid b, \sigma) \propto p(b \mid x, \sigma) p(x) \]

- Maximum-a-posteriori estimation:

\[ x_{MAP} = \arg\min_{x} \frac{1}{2\sigma^2} \|b - Ax\|_2^2 + \Gamma(x) \]

- Regularized least squares (use ADMM)
Scientific Sensors

- e.g., Andor iXon Ultra 897: cooled to -100°C
- Scientific CMOS & CCD
- Reduce pretty much all noise, except for photon noise
What is Photon (or Shot) Noise?

• Fluctuations in the light emitted by a source, i.e. due to particle nature of light (*emission*).

• Fluctuation when the incident light is converted to charge, i.e. due to variability in the number of electrons (*detection*).

• Same for re-emission → cascading Poisson processes are also described by a Poisson process [Teich and Saleh 1998].
Photon Noise

• Noise is signal dependent!

• For $N$ measured photo-electrons
  • standard deviation is $\sigma = \sqrt{N}$, variance is $\sigma^2 = N$
  • mean is $N$

• Poisson distribution
  $$f(k; N) = \frac{N^k e^{-N}}{k!}$$
Photon Noise - SNR

Signal-to-noise ratio
(mean / $\sigma$)

$$SNR = \frac{N}{\sqrt{N}}$$

$SNR$ in dB

N photons: $= \sqrt{N}$

2N photons: $= \sqrt{2\sqrt{N}}$

$SNR$ is nonlinear!
Photon Noise - SNR

signal-to-noise ratio

$$SNR = \frac{N}{\sqrt{N}}$$

N photons: $= \sqrt{N}$

2N photons: $= \sqrt{2\sqrt{N}}$

nonlinear!
Maximum Likelihood Solution for Poisson Noise

• Image formation: \( \mathbf{b} \sim \mathcal{P}(\mathbf{Ax}) \)

• Probability of measurement \( i \):
  \[
p(\mathbf{b}_i|\mathbf{x}) = \frac{(\mathbf{Ax})_i^{b_i} e^{-(\mathbf{Ax})_i}}{b_i!}
  \]

• Joint probability of all \( M \) measurements (use notation trick \( u^v = e^{\log(u)v} \)):
  \[
p(\mathbf{b}|\mathbf{x}) = \prod_{i=1}^{M} p(\mathbf{b}_i|\mathbf{x}) = \prod_{i=1}^{M} e^{\log((\mathbf{Ax})_i) b_i} \times e^{-(\mathbf{Ax})_i} \times \frac{1}{b_i!}
  \]
Maximum Likelihood Solution for Poisson Noise

• Log-likelihood function:

$$
\log (L(x)) = \log(p(b|x))
$$

$$
= \sum_{i=1}^{M} \log((Ax)_i) b_i - \sum_{i=1}^{M} (Ax)_i - \sum_{i=1}^{M} \log(b_i!)
$$

$$
= \log((Ax)^T b) - (Ax)^T 1 - \sum_{i=1}^{M} \log(b_i!)
$$

• Gradient:

$$
\nabla \log(L(x)) = A^T \text{diag}(Ax)^{-1} b - A^T 1 = A^T \left( \frac{b}{Ax} \right) - A^T 1
$$
Maximum Likelihood Solution for Poisson Noise

- Richardson-Lucy algorithm:
  iterative approach to ML estimation for Poisson noise

- Simple idea:
  1. At solution, gradient will be zero
  2. When converged, further iterations will not change, i.e.

\[
\frac{x^{(q+1)}}{x^{(q)}} = 1
\]
Richardson-Lucy Algorithm

• Equate gradient to zero

\[ \nabla \log (L(x)) = A^T \text{diag}(Ax)^{-1} b - A^T 1 = A^T \left( \frac{b}{Ax} \right) - A^T 1 = 0 \]

• Rearrange so that 1 is on one side of equation

\[ \text{diag} \left( A^T 1 \right)^{-1} A^T \text{diag}(Ax)^{-1} b = \frac{A^T \left( \frac{b}{Ax} \right)}{A^T 1} = 1 \]
Richardson-Lucy Algorithm

- Equate gradient to zero

$$\nabla \log (L(x)) = A^T \text{diag}(Ax)^{-1} b - A^T 1 = A^T \left( \frac{b}{Ax} \right) - A^T 1 = 0$$

- Rearrange so that 1 is on one side of equation

$$\text{diag}(A^T 1)^{-1} A^T \text{diag}(Ax)^{-1} b = \frac{A^T \left( \frac{b}{Ax} \right)}{A^T 1} = 1$$

- Set $x^{(q+1)}/x^{(q)} = 1$

$$x^{(q+1)} = \left( \text{diag}(A^T 1)^{-1} A^T \text{diag}(Ax)^{-1} b \right) \cdot x^{(q)} = \frac{A^T \left( \frac{b}{Ax} \right)}{A^T 1} \cdot x^{(q)}$$
Richardson-Lucy Algorithm

• For any multiplicative update rule scheme:
  • start with positive initial guess (e.g. random values)
  • apply iteration scheme
  • future updates are guaranteed to remain positive
  • always get smaller residual

• RL multiplicative update rules:

\[ x^{(q+1)} = \left( \text{diag} \left( A^T 1 \right)^{-1} A^T \text{diag} \left( Ax \right)^{-1} b \right) \cdot x^{(q)} = \frac{A^T \left( \frac{b}{Ax} \right)}{A^T 1} \cdot x^{(q)} \]
Richardson-Lucy Algorithm - Deconvolution

Blurry & Noisy Measurement

RL Deconvolution

\[ x^{(q+1)} = \left( \text{diag} \left( A^T 1 \right)^{-1} A^T \text{diag} \left( Ax \right)^{-1} b \right) \cdot x^{(q)} = \frac{A^T \left( \frac{b}{Ax} \right)}{A^T 1} \cdot x^{(q)} \]
Richardson-Lucy Algorithm

• What went wrong?

• Poisson deconvolution is a tough problem, without priors it’s pretty much hopeless

• Let’s try to incorporate the one prior we have learned: total variation
Richardson-Lucy Algorithm + TV

• Log-likelihood function:

$$\log (L_{TV}(x)) = \log (p(b|x)) + \log (p(x)) = \log (A^Tx)^T b - (A^x)^T 1 - \sum_{i=1}^{M} \log (b_i!) - \lambda \|Dx\|_1$$

• Gradient:

$$\nabla \log (L_{TV}(x)) = A^T \text{diag}(Ax)^{-1} b - A^T 1 + \nabla \lambda \|Dx\|_1 = A^T \left( \frac{b}{Ax} \right) - A^T 1 - \nabla \lambda \|Dx\|_1$$
Richardson-Lucy Algorithm + TV

- Gradient of anisotropic TV term

\[
\nabla \lambda \|Dx\|_1 = \lambda \nabla \sum_{ij} \left( \sqrt{(\nabla x_i x_j)^2} + \sqrt{(\nabla y_i x_j)^2} \right) = -\lambda \left( \frac{D_{x_i x_j}}{|D_{x_i x_j}|} + \frac{D_{y_i x_j}}{|D_{y_i x_j}|} \right)
\]

- This is “dirty”: possible division by 0!
Richardson-Lucy Algorithm + TV

- Follow the same logic as RL, RL+TV multiplicative update rules:

$$x^{(q+1)} = \frac{A^T (\frac{b}{A^T x})}{A^T 1 - \lambda \left( \frac{D_{xx} x}{|D_{xx}|} + \frac{D_{yy} x}{|D_{yy}|} \right)} \cdot x^{(q)}$$

- 2 major problems & solution “hacks”:
  1. still possible division by 0 when gradient is zero
     - set fraction to 0 if gradient is 0
  2. multiplicative update may become negative!
     - only work with (very) small values for lambda
(A dirty but easy approach to) Richardson-Lucy with a TV Prior

Measurements

Log Residual

Mean Squared Error
Signal-to-Noise Ratio (SNR)

\[
SNR = \frac{\text{mean pixel value}}{\text{standard deviation of pixel value}} = \frac{PQ_e t}{\sqrt{PQ_e t + Dt + N_r^2}}
\]

- \(P\) = incident photon flux (photons/pixel/sec)
- \(Q_e\) = quantum efficiency
- \(t\) = exposure time (sec)
- \(D\) = dark current (electrons/pixel/sec), including hot pixels
- \(N_r\) = read noise (rms electrons/pixel), including fixed pattern noise
Signal-to-Noise Ratio (SNR)

\[ SNR = \frac{\text{mean pixel value}}{\text{standard deviation of pixel value}} = \frac{PQ_e t}{\sqrt{PQ_e t + Dt + N_r^2}} \]

- \( P \): incident photon flux (photons/pixel/sec)
- \( Q_e \): quantum efficiency
- \( t \): exposure time (sec)
- \( D \): dark current (electrons/pixel/sec), including hot pixels
- \( N_r \): read noise (rms electrons/pixel), including fixed pattern noise
Next: Compressive Imaging

- Single pixel camera
- Compressive hyperspectral imaging
- Compressive light field imaging

Wakin et al. 2006
References and Further Reading


- Please read the lecture notes, especially for the “clean” ADMM derivation for solving the maximum likelihood estimation of Poisson reconstruction with TV prior!