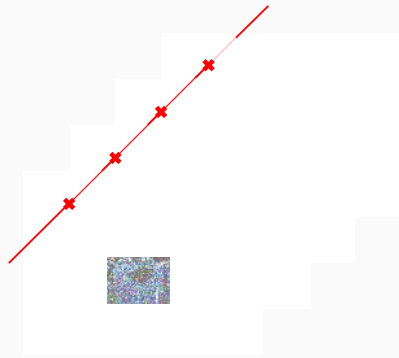


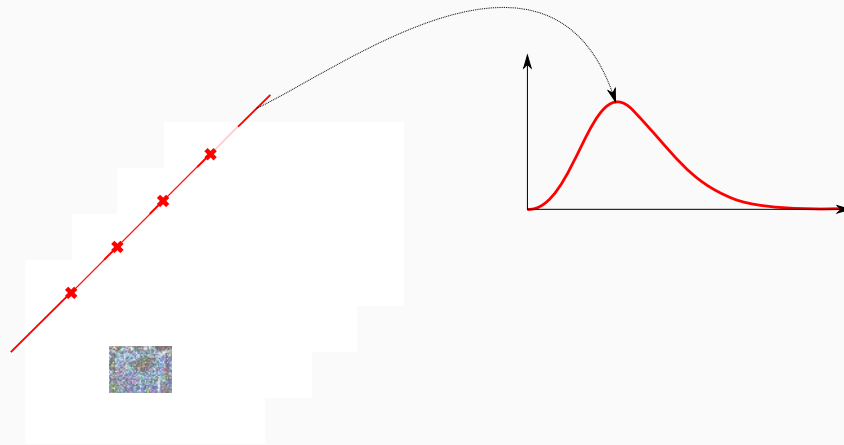
## Digital optical imagery – Noise modeling

- Take several pictures of the same scene, and focus on one given pixel,
- There are always unwanted fluctuations around the “true” pixel value,
- These fluctuations are called noise,
- Usually described by a probability density or mass function (pdf/pmf),
- Stochastic process  $Y$  parametrized by a deterministic signal of interest  $x$ .



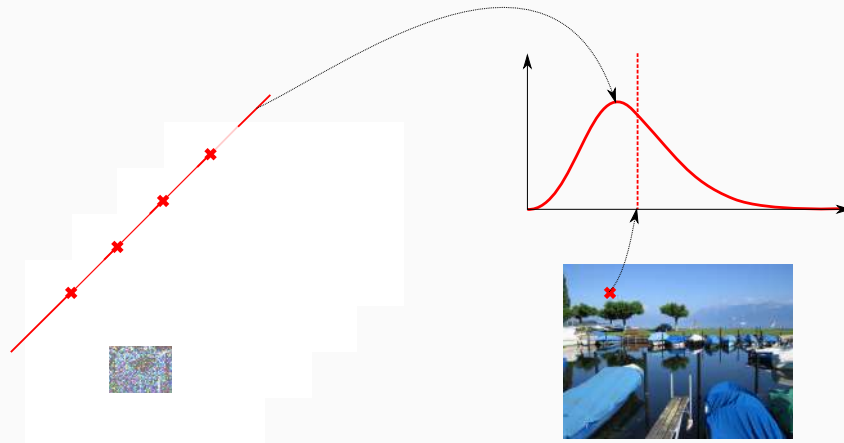
## Digital optical imagery – Noise modeling

- Take several pictures of the same scene, and focus on one given pixel,
- There are always unwanted fluctuations around the “true” pixel value,
- These fluctuations are called noise,
- Usually described by a probability density or mass function (pdf/pmf),
- Stochastic process  $Y$  parametrized by a deterministic signal of interest  $x$ .



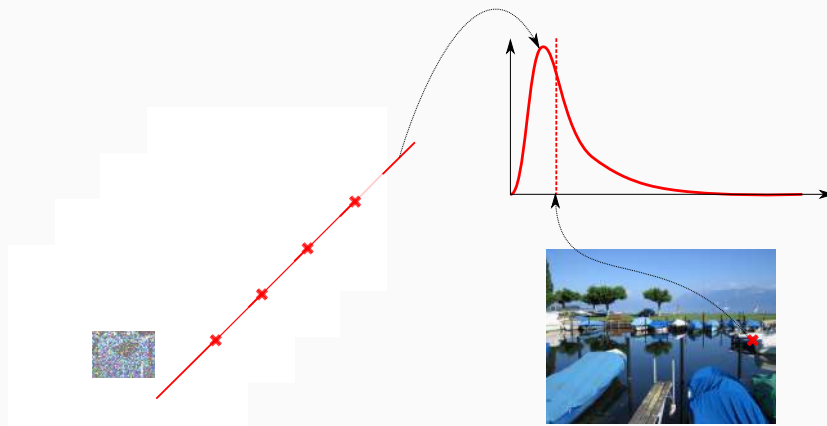
## Digital optical imagery – Noise modeling

- Take several pictures of the same scene, and focus on one given pixel,
- There are always unwanted fluctuations around the “true” pixel value,
- These fluctuations are called noise,
- Usually described by a probability density or mass function (pdf/pmf),
- Stochastic process  $Y$  parametrized by a deterministic signal of interest  $x$ .



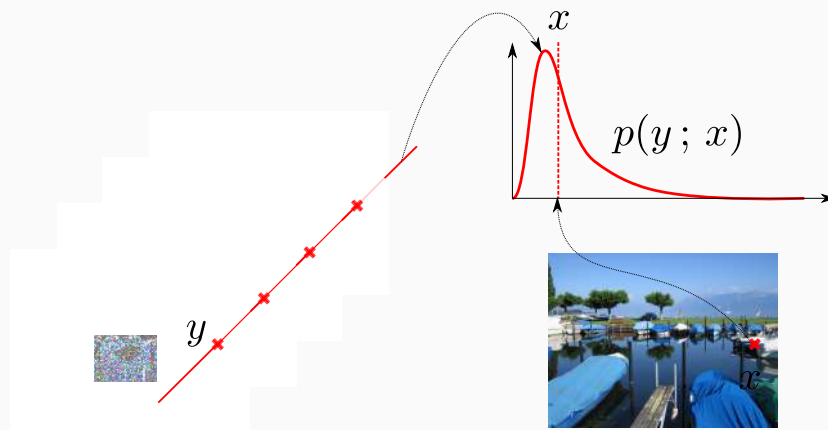
## Digital optical imagery – Noise modeling

- Take several pictures of the same scene, and focus on one given pixel,
- There are always unwanted fluctuations around the “true” pixel value,
- These fluctuations are called noise,
- Usually described by a probability density or mass function (pdf/pmf),
- Stochastic process  $Y$  parametrized by a deterministic signal of interest  $x$ .



## Digital optical imagery – Noise modeling

- Take several pictures of the same scene, and focus on one given pixel,
- There are always unwanted fluctuations around the “true” pixel value,
- These fluctuations are called noise,
- Usually described by a probability density or mass function (pdf/pmf),
- Stochastic process  $Y$  parametrized by a deterministic signal of interest  $x$ .



$x$  true unknown pixel value,  $y$  noisy observed value (a realization of  $Y$ ),  
link:  $p_Y(y; x)$  noise model

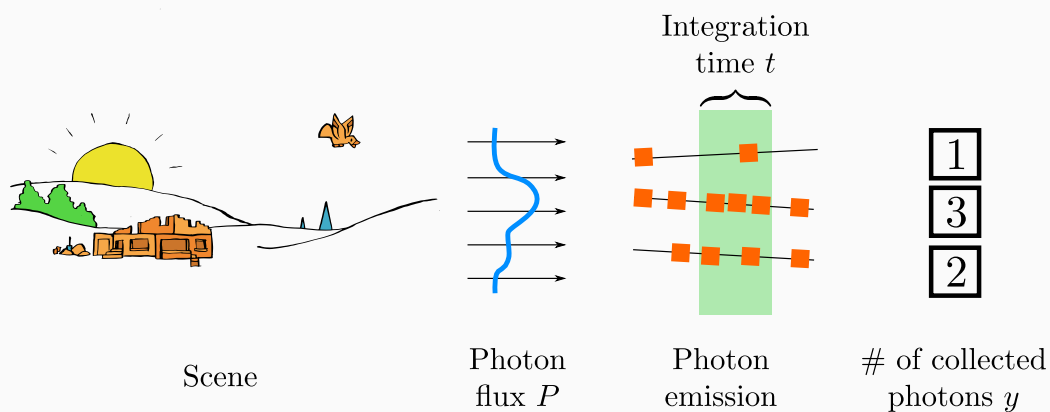
## Digital optical imagery – Shot noise

### Shot noise

- Number of captured photons  $y \in \mathbb{N}$  fluctuates around the signal of interest

$$x = PQ_e t$$

- $x$ : expected quantity of light
  - $Q_e$ : quantum efficiency (depends on wavelength)
  - $P$ : photon flux (depends on light intensity and pixel size)
  - $t$ : integration time
- Variations depends on exposure times and light conditions.



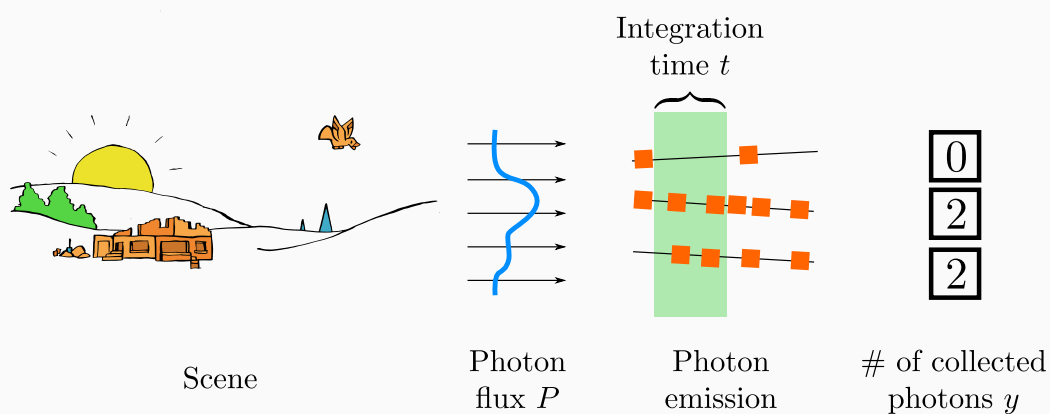
## Digital optical imagery – Shot noise

### Shot noise

- Number of captured photons  $y \in \mathbb{N}$  fluctuates around the signal of interest

$$x = PQ_e t$$

- $x$ : expected quantity of light
  - $Q_e$ : quantum efficiency (depends on wavelength)
  - $P$ : photon flux (depends on light intensity and pixel size)
  - $t$ : integration time
- Variations depends on exposure times and light conditions.



## Digital optical imagery – Shot noise

### Shot noise and Poisson distribution

- Distribution of  $Y$  modeled by the Poisson distribution

$$p_Y(y; x) = \frac{x^y e^{-x}}{y!}$$

- Number of photons  $y \in \mathbb{N}$  fluctuates around the signal of interest  $x \in \mathbb{R}$

$$\mathbb{E}[Y] = \sum_{y=0}^{\infty} y p_Y(y; x) = x$$

- Fluctuations proportional to  $\text{Std}[Y] = \sqrt{\text{Var}[Y]} = \sqrt{x}$

$$\text{Var}[Y] = \sum_{y=0}^{\infty} (y - x)^2 p_Y(y; x) = x$$

- Inherent when counting particles in a given time window

We write  $Y \sim \mathcal{P}(x)$



## Digital optical imagery – Shot noise

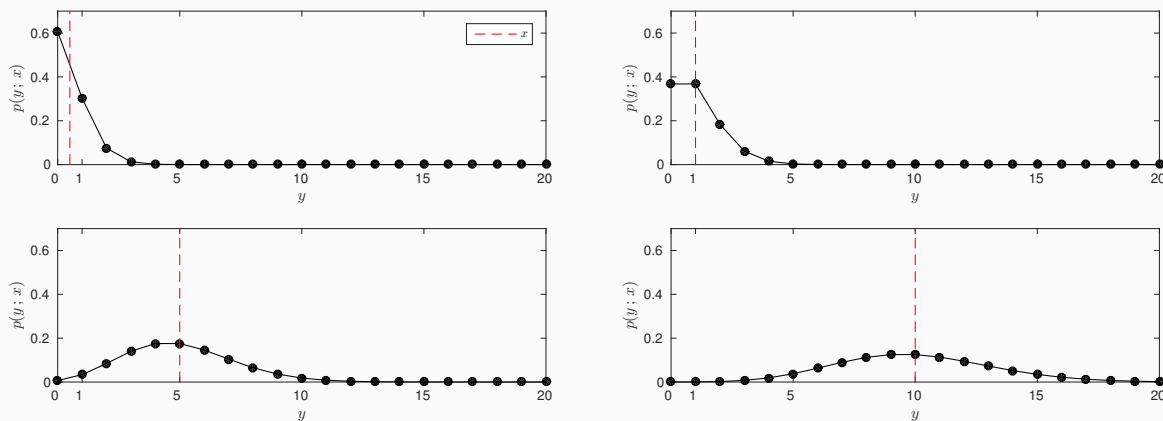


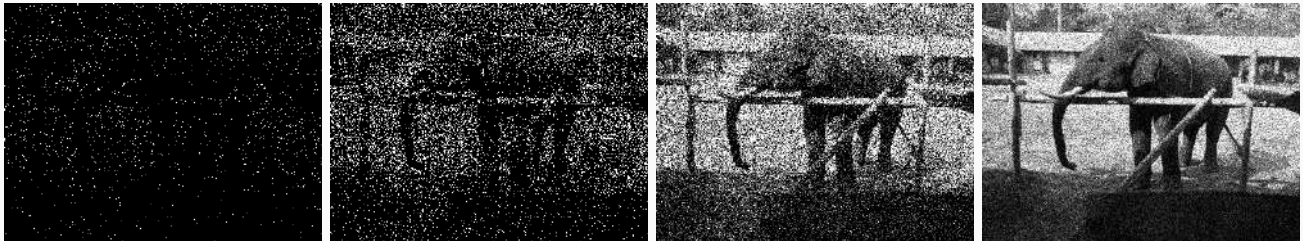
Figure 1 – Distribution of  $Y$  for a given quantity of light  $x$

- For  $x = 0.5$ : mostly 0 photons, Spread  $\approx 0.7$
- For  $x = 1$ : mostly 0 or 1 photons, Spread = 1
- For  $x \gg 1$ : bell shape around  $x$ , Spread =  $\sqrt{x}$

Spread is higher when  $x = PQ_e t$  is large.

Should we prefer small exposure time  $t$ ? and lower light conditions  $P$ ?

## Digital optical imagery – Shot noise



(a) Peak = 0.05

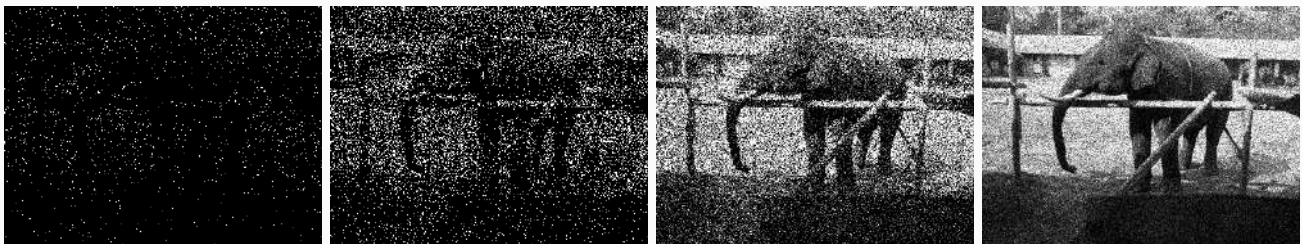
(b) Peak = 0.40

(c) Peak = 3.14

(d) Peak = 24.37

**Figure 2** – Aspect of shot noise under different light conditions. Peak =  $\max_i x_i$ .

## Digital optical imagery – Shot noise



(a) Peak = 0.05

(b) Peak = 0.40

(c) Peak = 3.14

(d) Peak = 24.37

**Figure 2** – Aspect of shot noise under different light conditions. Peak =  $\max_i x_i$ .

### Signal to Noise Ratio

$$\text{SNR} = \frac{x}{\sqrt{\text{Var}[Y]}}, \quad \text{for shot noise} \quad \text{SNR} = \sqrt{x}$$

- Measure of difficulty/quality
- The higher the easier/better
- Rose criterion: an SNR of at least 5 is needed to be able to distinguish image features at 100% certainty.

The spread (variance) is not informative,  
what matters is the spread relatively to the signal (SNR)

## Digital optical imagery – Readout noise

### Readout noise (a.k.a, electronic noise)

- Inherent to the process of converting CCD charges into voltage
- Measures  $y \in \mathbb{R}$  fluctuate around a voltage  $x \in \mathbb{R}$

$$\mathbb{E}[Y] = \int y p_Y(y; x) dy = x$$

- Fluctuations are independent of  $x$

$$\text{Var}[Y] = \int (y - x)^2 p_Y(y; x) dy = \sigma^2$$

- Described as Gaussian distributed

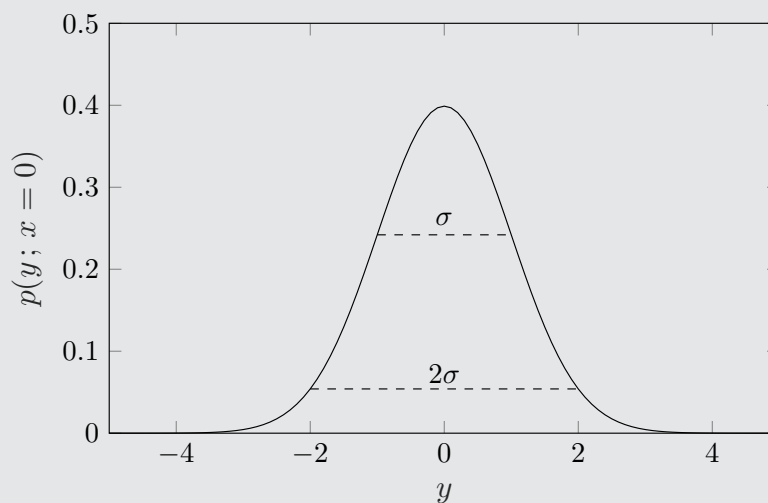
$$p_Y(y; x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y - x)^2}{2\sigma^2}\right)$$

- Additive behavior:  $Y = x + W$ ,  $W \sim \mathcal{N}(0, \sigma^2)$

We write  $Y \sim \mathcal{N}(x, \sigma^2)$

## Digital optical imagery – Readout noise

### Gaussian/Normal distribution



- Symmetric with bell shape.
- Common to models  $\pm\sigma$  uncertainties with very few outliers  
 $\mathbb{P}[|Y - x| \leq \sigma] \approx 0.68$ ,  $\mathbb{P}[|Y - x| \leq 2\sigma] \approx 0.95$ ,  $\mathbb{P}[|Y - x| \leq 3\sigma] \approx 0.99$ .
- Arises in many problems due to the Central Limit Theorem.
- Simple to manipulate: ease computation in many cases.

## Digital optical imagery – Shot noise vs Readout noise

Shot noise is signal-dependent (Poisson noise)



Readout noise is signal-independent (Gaussian noise)



## Digital optical imagery – Thermal and total noise

### Thermal noise (a.k.a, dark noise)

- Number of generated electrons fluctuates with the CCD temperature
- Additive Poisson distributed:  $Y = x + N$  with  $N \sim \mathcal{P}(\lambda)$
- Signal independent

## Digital optical imagery – Thermal and total noise

### Thermal noise (a.k.a, dark noise)

- Number of generated electrons fluctuates with the CCD temperature
- Additive Poisson distributed:  $Y = x + N$  with  $N \sim \mathcal{P}(\lambda)$
- Signal independent

### Total noise in CCD models

$$Y = Z + N + W$$

$$\text{with } \begin{cases} Z \sim \mathcal{P}(x), \\ N \sim \mathcal{P}(\lambda), \\ W \sim \mathcal{N}(0, \sigma^2). \end{cases}$$

$$\text{SNR} = \frac{x}{\sqrt{x + \lambda + \sigma^2}}$$

$$\text{where } x = PQ_e t, \quad \lambda = Dt$$

- $t$ : exposure time
- $P$ : photon flux per pixel  
(depends on luminosity)
- $Q_e$ : quantum efficiency  
(depends on wavelength)
- $D$ : dark current  
(depends on temperature)
- $\sigma$ : readout noise  
(depends on electronic design)



## Digital optical imagery – How to reduce noise?

$$\text{SNR} = \frac{x}{\sqrt{x + \lambda + \sigma^2}} \quad \text{where} \quad x = PQ_e t, \quad \lambda = Dt$$

### Photon noise

- Cannot be reduced via camera design
- Reduced by using a longer exposure time  $t$
- Reduced by increasing the scene luminosity, higher  $P$  (e.g., using a flash)
- Reduced by increasing the aperture, higher  $P$

## Digital optical imagery – How to reduce noise?

$$\text{SNR} = \frac{x}{\sqrt{x + \lambda + \sigma^2}} \quad \text{where} \quad x = PQ_e t, \quad \lambda = Dt$$

### Photon noise

- Cannot be reduced via camera design
- Reduced by using a longer exposure time  $t$
- Reduced by increasing the scene luminosity, higher  $P$  (e.g., using a flash)
- Reduced by increasing the aperture, higher  $P$

### Thermal noise

- Reduced by cooling the CCD, *i.e.*, lower  $D \Rightarrow$  More expensive cameras
- Or by using a longer exposure time  $t$

## Digital optical imagery – How to reduce noise?

$$\text{SNR} = \frac{x}{\sqrt{x + \lambda + \sigma^2}} \quad \text{where} \quad x = PQ_e t, \quad \lambda = Dt$$

### Photon noise

- Cannot be reduced via camera design
- Reduced by using a longer exposure time  $t$
- Reduced by increasing the scene luminosity, higher  $P$  (e.g., using a flash)
- Reduced by increasing the aperture, higher  $P$

### Thermal noise

- Reduced by cooling the CCD, *i.e.*, lower  $D \Rightarrow$  More expensive cameras
- Or by using a longer exposure time  $t$

### Readout noise

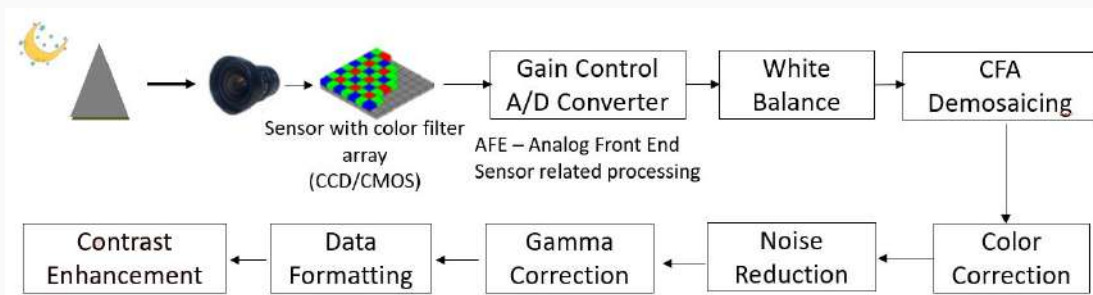
- Reduced by employing carefully designed electronics, *i.e.*, lower  $\sigma \Rightarrow$  More expensive cameras

Or, reduced by image restoration software. But only if such models are accurate

## Digital optical imagery – Are these models accurate?

### Processing pipeline

- There are always some pre-processing steps such as
  - white balance: to make sure neutral colors appear neutral,
  - demosaicing: to create a color image from incomplete color samples,
  - $\gamma$ -correction: to optimize the usage of bits, and fit human perception of brightness,
  - compression: to improve memory usage (e.g., JPEG).
- Technical details often hidden by the camera vendors
- The noise in the resulting becomes much harder to model



Source: Y. Gong and Y. Lee

## Digital optical imagery – Noise models and post-processing

### Example ( $\gamma$ -correction)

$$y^{(\text{new})} = Ay^\gamma$$



(a) Non corrected



(b)  $\gamma$ -corrected



(c) Zoom  $\times 8$



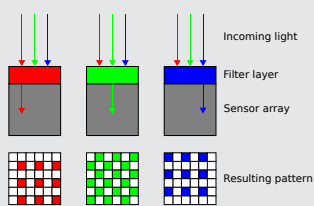
(d) Zoom  $\times 30$

What is the influence on the noise?

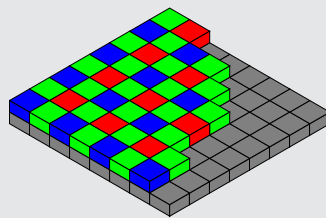
- CCD model:  $\mathbb{E}[Y] = x + \lambda$  and  $\text{Var}[Y] = x + \lambda + \sigma^2$
- Delta method:  $\text{Var}[f(Y)] \approx f'(\mathbb{E}[Y])^2 \text{Var}[Y]$
- Resulting model:  $\text{Var}[Y^{(\text{new})}] \approx A^2 \gamma^2 (x + \lambda)^{2(\gamma-1)} (x + \lambda + \sigma^2)$
- But  $A$  and  $\gamma$  are usually not known ☹

## Digital optical imagery – Noise models and post-processing

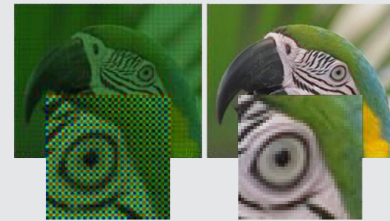
### Example (Demosaicing)



(a) Bayer filter



(b) Bayer pattern



(c) Demosaicing

Basic idea:

- Use interpolation techniques.
- Bilinear interpolation: the red value of a non-red pixel is computed as the average of the two or four adjacent red pixels, and similarly for blue and green.

What is the influence on the noise?

- noise is no longer independent from one pixel to another,
- noise becomes spatially correlated.

Compression also create spatial correlations.

## Digital optical imagery – Noise models and correlations

### Reminder of basic statistics

- $X$  and  $Y$  two real random variables
- Independence:  $p_{X,Y}(x, y) = p_X(x)p_Y(y)$
- Decorrelation:  $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]$
- Covariance:  $\text{Cov}(X, Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])]$   
 $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y] + \text{Cov}(X, Y)$   
 $\text{Var}(X) = \text{Cov}(X, X)$
- Correlation:  $\text{Corr}(X, Y) = \frac{\text{Cov}(X, Y)}{\sqrt{\text{Var}[X]\text{Var}[Y]}}$   
 $\text{Corr}(X, X) = 1$

①	Independence	$\Leftrightarrow / \Rightarrow / \Leftarrow$	Decorrelation	?
②	$\text{Corr}(X, Y) = 1$	$\Leftrightarrow / \Rightarrow / \Leftarrow$	$X = Y$	?
③	$\text{Corr}(X, Y) = -1$	$\Leftrightarrow / \Rightarrow / \Leftarrow$	$X = aY + b, a < 0$	?

## Digital optical imagery – Noise models and correlations

### Reminder of multivariate statistics

- $X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix}$  and  $Y = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_m \end{pmatrix}$  two real random vectors
- Entries are independent:  $p_X(x) = \prod_k p_{X_k}(x_k)$
- Covariance matrix:  $\text{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])(X - \mathbb{E}[X])^T] \in \mathbb{R}^{n \times n}$   
 $\text{Var}(X)_{ij} = \text{Cov}(X_i, X_j)$
- Correlation matrix  $\text{Corr}(X)_{ij} = \text{Corr}(X_i, X_j)$
- Cross-covariance matrix:  $\text{Cov}(X, Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])^T] \in \mathbb{R}^{m \times n}$
- Cross-correlation matrix:  $\text{Corr}(X, Y)_{ij} = \text{Corr}(X_i, Y_j)$

NB: cross-correlation definition is slightly different in signal processing (in few slides)



## Digital optical imagery – Noise models and correlations

- See an image  $x$  as a vector of  $\mathbb{R}^n$ ,
- Its observation  $y$  is a realization of a random vector

$$Y = x + W \quad \text{and} \quad p_Y(y; x) = p_W(y - x; x),$$

- In general, noise is assumed to be zero-mean  $\mathbb{E}[W] = 0$ , then

$$\mathbb{E}[Y] = x \quad \text{and} \quad \text{Var}[Y] = \text{Var}[W] = \mathbb{E}[WW^T] = \Sigma.$$

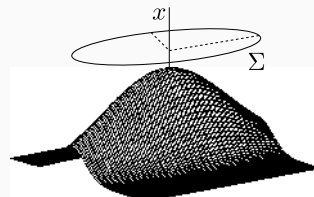
- $\Sigma$  encodes variances and correlations (may depend on  $x$ ).
- $p_Y$  is often modeled with a multivariate Gaussian/normal distribution

$$p_Y(y; x) \approx \frac{1}{\sqrt{2\pi}^n |\Sigma|^{1/2}} \exp \left( -\frac{1}{2} (y - x)^T \Sigma^{-1} (y - x) \right).$$



Underlying noise distribution

$\approx$



Gaussian approximation  $Y \sim \mathcal{N}(x; \Sigma)$

## Digital optical imagery – Noise models and correlations

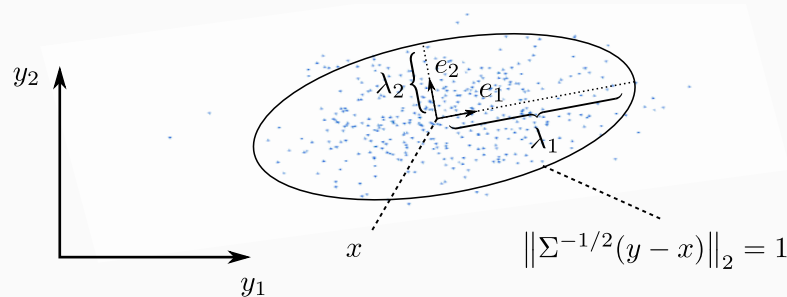
### Properties of covariance matrices

- $\Sigma = \text{Var}[Y]$  is square, symmetric and non-negative definite:

$$x^T \Sigma x \geq 0, \quad \text{for all } x \neq 0 \text{ (eigenvalues } \lambda_i \geq 0).$$

- If all  $Y_k$  are linearly independent, then

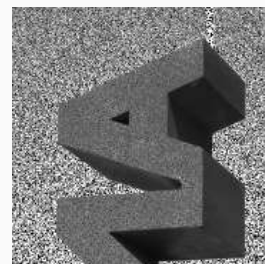
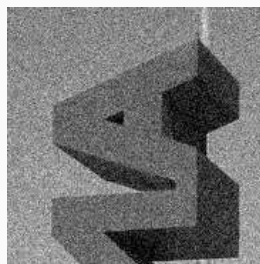
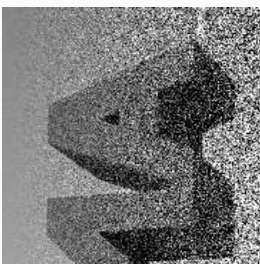
- $\Sigma$  is positive definite:  $x^T \Sigma x > 0$ , for all  $x \neq 0$  ( $\lambda_i > 0$ ),
- $\Sigma$  is invertible and  $\Sigma^{-1}$  is also symmetric positive definite,
- Mahanalobis distance:  $\sqrt{(y - x)^T \Sigma^{-1} (y - x)} = \|\Sigma^{-1/2} (y - x)\|_2$ ,
- Its isoline  $\{x ; \|\Sigma^{-1/2} (y - x)\|_2 = c, c > 0\}$  describes an ellipsoid of center  $x$  and semi-axes the eigenvectors  $e_i$  with length  $c\lambda_i$ .



## Digital optical imagery – Noise dictionary

### Vocabulary in signal processing

- White noise: zero-mean noise + no correlations
- Stationary noise: identically distributed whatever the location
- Colored noise: stationary with pixels influencing their neighborhood
- Signal dependent: noise statistics depends on the signal intensity
- Space dependent: noise statistics depends on the location
- AWGN: Additive White Gaussian Noise:  $Y \sim \mathcal{N}(x; \sigma^2 \text{Id}_n)$



## Digital optical imagery – Noise models and correlations

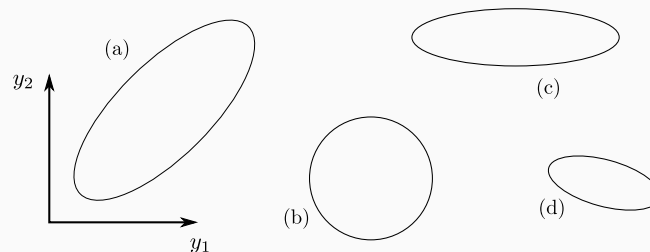
### How is it encoded in $\Sigma$ ?

- ①  $\Sigma$  diagonal: noise is uncorrelated – *white*
- ②  $\Sigma_{i,i} = f(s_i)$ : variance depends on pixel location  $s_i$  – *space dependent*
- ③  $\Sigma_{i,i} = f(x_i)$ : variance depends on pixel value  $x_i$  – *signal dependent*
- ④  $\Sigma_{i,j} = f(s_i - s_j)$ : correlations depends on the shift – *stationary*

For 1d signals,  $\Sigma$  is Toeplitz:  $\Sigma = \begin{pmatrix} a & b & \dots & c \\ d & a & \ddots & \vdots \\ \vdots & \ddots & \ddots & b \\ e & \dots & d & a \end{pmatrix}$

⑤  $\Sigma = \underbrace{\begin{pmatrix} \sigma^2 & & 0 \\ & \ddots & \\ 0 & & \sigma^2 \end{pmatrix}}_{=\sigma^2 \text{Id}_n}$ : noise is homoscedastic  
( $\neq$  heteroscedastic)

– *white+stationary*



## Digital optical imagery – Settings to avoid noise



(a) Very short exposure



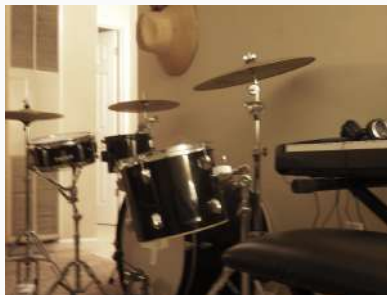
(b) Short exposure



(c) Flash



(d) Normal exposure



(e) Long exposure



(f) Long + hand shaking

- Short exposure: too much noise
- Using a flash: change the aspect of the scene
- Long exposure: subject to **blur** and saturation (use a tripod)

## What is blur?

---



Blur: The best of, 2000

## Digital optical imagery – Blur

### Motion blur

- Moving object
- Camera shake
- Atmospheric turbulence
- Long exposure time



### Camera blur

- Limited resolution
- Diffraction
- Bad focus
- Wrong optical design



### Bokeh

- Out-of-focus parts
- Often for artistic purpose



## Digital optical imagery – Blur

### Motion blur

- Moving object
- Camera shake
- Atmospheric turbulence
- Long exposure time



### Camera blur

- Limited resolution
- Diffraction
- Bad focus
- Wrong optical design



### Bokeh

- Out-of-focus parts
- Often for artistic purpose



How to model blur?



## Digital optical imagery – Blur – Linear property

$$\begin{aligned} \text{Blur} \left( \begin{array}{c} \text{Image 1} \\ \text{Image 2} \end{array} \right) &= \text{Blur} \left( \begin{array}{c} \text{Image 1} \\ + \\ \text{Image 2} \end{array} \right) \\ &= \text{Blur} \left( \begin{array}{c} \text{Image 1} \end{array} \right) + \text{Blur} \left( \begin{array}{c} \text{Image 2} \end{array} \right) \\ &= \begin{array}{c} \text{Blurred Image 1} \\ + \\ \text{Blurred Image 2} \end{array} = \begin{array}{c} \text{Blurred Image 1+2} \end{array} \end{aligned}$$

Blur is linear

## Digital optical imagery – Linear blur

### Linear model of blur

- Observed pixel values are a mixture of the underlying ones

$$y_{i,j} = \sum_{k=1}^n \sum_{l=1}^n h_{i,j,k,l} x_{k,l} \quad \text{where} \quad h_{k,l} \geq 0 \quad \text{and} \quad \sum_{l=1}^n h_{k,l} = 1$$

- Matrix/vector representation:  $y = Hx$        $y \in \mathbb{R}^n, x \in \mathbb{R}^n, H \in \mathbb{R}^{n \times n}$

## Digital optical imagery – Linear blur

### Linear model of blur

- Observed pixel values are a mixture of the underlying ones

$$y_{i,j} = \sum_{k=1}^n \sum_{l=1}^n h_{i,j,k,l} x_{k,l} \quad \text{where} \quad h_{k,l} \geq 0 \quad \text{and} \quad \sum_{l=1}^n h_{k,l} = 1$$

- Matrix/vector representation:  $y = Hx$        $y \in \mathbb{R}^n$ ,  $x \in \mathbb{R}^n$ ,  $H \in \mathbb{R}^{n \times n}$

$$y = \begin{pmatrix} \overbrace{h_{1,1,1,1} \quad \dots \quad h_{1,1,1,n_2}}^{\text{First line}} & \dots & \overbrace{h_{1,1,n_1,1} \quad \dots \quad h_{1,1,n_1,n_2}}^{\text{Last line}} \\ \vdots & & \vdots \\ h_{1,n_2,1,1} \quad \dots \quad h_{1,n_2,1,n_2} & \dots & h_{1,n_2,n_1,1} \quad \dots \quad h_{1,n_2,n_1,n_2} \\ \hline \vdots & & \vdots \\ \hline h_{n_1,1,1,1} \quad \dots \quad h_{n_1,1,1,n_2} & \dots & h_{n_1,1,n_1,1} \quad \dots \quad h_{n_1,1,n_1,n_2} \\ \vdots & & \vdots \\ h_{n_1,n_2,1,1} \quad \dots \quad h_{n_1,n_2,1,n_2} & \dots & h_{n_1,n_2,n_1,1} \quad \dots \quad h_{n_1,n_2,n_1,n_2} \end{pmatrix} \begin{pmatrix} x_{1,1} \\ \vdots \\ x_{1,n_2} \\ \hline \vdots \\ \hline x_{n_1,1} \\ \vdots \\ x_{n_1,n_2} \end{pmatrix}$$

## Digital optical imagery – Point Spread Function (PSF)

$$\begin{array}{c}
 H \times \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}}_{\text{Only one 1 for some pixel } (i, j)} = \underbrace{\begin{pmatrix} h_{1,1,i,j} \\ \vdots \\ h_{i,j-1,i,j} \\ h_{i,j,i,j} \\ h_{i,j+1,i,j} \\ \vdots \\ h_{n_1,n_2,i,j} \end{pmatrix}}_{\text{One column of } H} \xrightarrow{\text{"reshape"}} \underbrace{\begin{pmatrix} h_{1,1,i,j} & h_{2,1,i,j} & \dots & h_{n_1,1,i,j} \\ \vdots & \vdots & & \vdots \\ h_{1,n_2,i,j} & h_{2,n_2,i,j} & \dots & h_{n_1,n_2,i,j} \end{pmatrix}}_{\substack{\text{System's impulse response at location } (i, j) \\ \text{called, Point spread function}}}
 \end{array}$$

$$\text{Blur} \left( \begin{pmatrix} \text{Grid with one solid black pixel} \end{pmatrix} \right) = \text{Grid}$$

## Digital optical imagery – Point Spread Function (PSF)

### Spatially varying PSF – non-stationary blur



## Digital optical imagery – Stationary blur

### Stationary blur

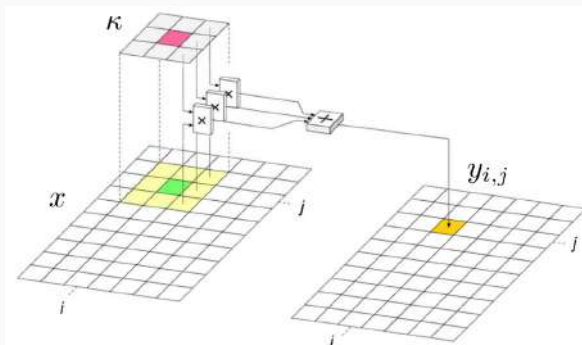
- Shift invariant: blurring depends only on the relative position:

$$h_{i,j,k,l} = \kappa_{k-i,l-j},$$

*i.e.*, same PSF everywhere.

- Corresponds to the (discrete) cross-correlation *(not the same as in statistics)*

$$y = \kappa \star x \quad \Leftrightarrow \quad y_{i,j} = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} \kappa_{k,l} x_{i+k,j+l}$$



## Digital optical imagery – Stationary blur

### Stationary blur

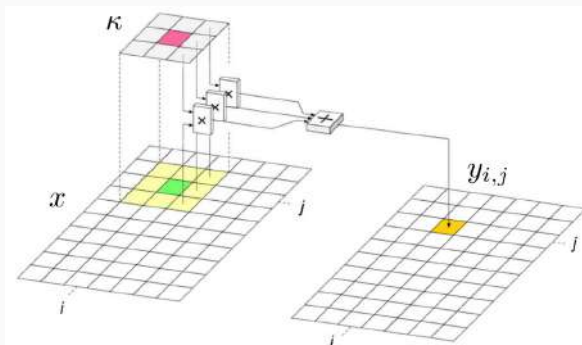
- Shift invariant: blurring depends only on the relative position:

$$h_{i,j,k,l} = \kappa_{k-i,l-j},$$

*i.e.*, same PSF everywhere.

- Corresponds to the (discrete) cross-correlation *(not the same as in statistics)*

$$y = \kappa \star x \Leftrightarrow y_{i,j} = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} \kappa_{k,l} x_{i+k,j+l}$$



Here  $\kappa$  has a  $q = 3 \times 3$  support

$$\Rightarrow \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} \equiv \sum_{k=-1}^{+1} \sum_{l=-1}^{+1}$$

$q$  called window size.

Direct computation requires

$$O(nq).$$

$$\Rightarrow q \ll n$$

## Digital optical imagery – Stationary blur

### Cross-correlation vs Convolution product

- If  $\kappa$  is complex then the cross-correlation becomes

$$y = \kappa \star x \quad \Leftrightarrow \quad y_{i,j} = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} \kappa_{k,l}^* x_{i+k,j+l}.$$

- Complex conjugate:  $(a + ib)^* = a - ib$ .
- $y = \kappa \star x$  can be re-written as the (discrete) convolution product

$$y = \nu * x \quad \Leftrightarrow \quad y_{i,j} = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} \nu_{k,l} x_{i-k,j-l} \quad \text{with} \quad \nu_{k,l} = \kappa_{-k,-l}^*.$$

- $\nu$  called convolution kernel.

Why convolution instead of cross-correlation?

- **Associative:**  $(f * g) * h = f * (g * h)$
- **Commutative:**  $f * g = g * f$

For cross-correlation, only true if the signal is Hermitian, i.e., if  $f_{k,l} = f_{-k,-l}^*$ .



## Digital optical imagery – Stationary blur

### $3 \times 3$ box convolution

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0								

Source: Steven Seitz

## Digital optical imagery – Stationary blur

### $3 \times 3$ box convolution

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0	10							

Source: Steven Seitz

## Digital optical imagery – Stationary blur

### $3 \times 3$ box convolution

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0	10	20						

Source: Steven Seitz

## Digital optical imagery – Stationary blur

### $3 \times 3$ box convolution

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0	10	20	30					

Source: Steven Seitz

## Digital optical imagery – Stationary blur

### $3 \times 3$ box convolution

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0	10	20	30	30				

Source: Steven Seitz

## Digital optical imagery – Stationary blur

### $3 \times 3$ box convolution

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	60	90	90	90	60	30	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	10	20	30	30	30	30	20	10	
	10	10	10	0	0	0	0	0	

Source: Steven Seitz

## Digital optical imagery – Convolution kernels

### Classical kernels

- Box kernel:

$$\kappa_{i,j} = \frac{1}{Z} \begin{cases} 1 & \text{if } \max(|i|, |j|) \leq \tau \\ 0 & \text{otherwise} \end{cases}$$

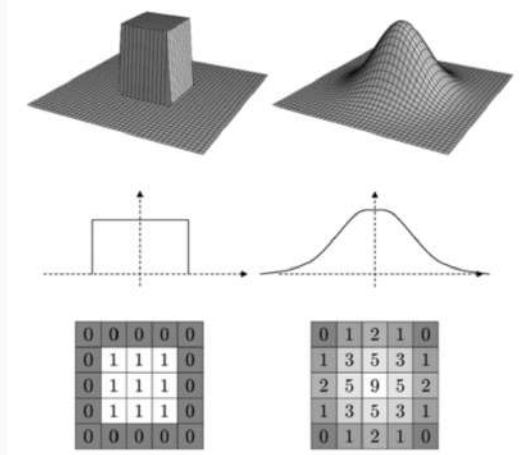
- Gaussian kernel:

$$\kappa_{i,j} = \frac{1}{Z} \exp\left(-\frac{i^2 + j^2}{2\tau^2}\right)$$

- Exponential kernel:

$$\kappa_{i,j} = \frac{1}{Z} \exp\left(-\frac{\sqrt{i^2 + j^2}}{\tau}\right)$$

- $Z$  normalization constant:  $Z = \sum_{i,j} \kappa_{i,j}$



## Digital optical imagery – Gaussian kernel

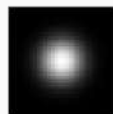
$$\kappa_{i,j} = \frac{1}{Z} \exp \left( -\frac{i^2 + j^2}{2\tau^2} \right)$$

### Influence of $\tau$

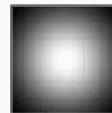
- $\sqrt{i^2 + j^2}$ : distance to the central pixel,
- $\tau$ : controls the influence of neighbor pixels, *i.e.*, the strength of the blur



Small  $\tau$



Medium  $\tau$

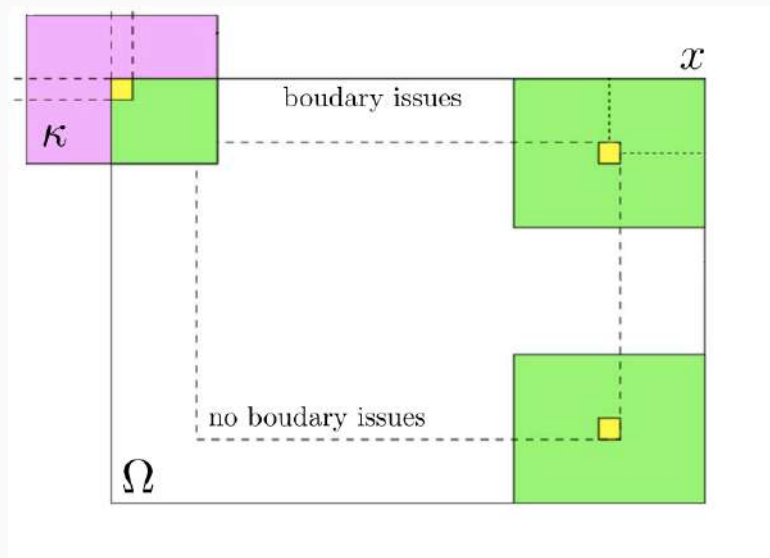


Large  $\tau$



## Digital optical imagery – Boundary conditions

How to deal when the kernel window overlaps outside the image domain?



i.e., how to evaluate  $y_{i,j} = \sum_{k,l} \kappa_{k,l} x_{i+k,j+l}$  when  $(i+k, j+l) \notin \Omega$ ?

## Digital optical imagery – Boundary conditions

Standard techniques:



zero-padding



extension



mirror



periodical