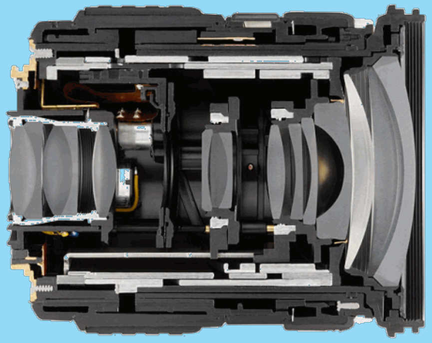
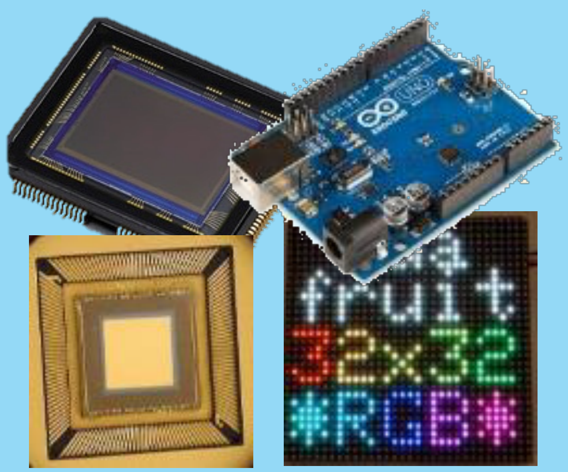


Optics

+



Sensors
&
devices

+

Signal
processing
&
algorithms

COMPUTATIONAL METHODS FOR IMAGING (AND VISION)

LECTURE 16: LSI IMAGING SYSTEMS (MATRIX-VECTOR FORMS)

PROF. JOHN MURRAY-BRUCE

LITTLE TYPO IN Q2

- **Question 2 (f) (ii)**
  - Should be $\Delta = \delta = 5\mu m$

(f) Plot the system transfer functions with the following parameters: $\lambda = 0.5\ \mu\text{m}$, $N = 1000$.

(i) Keep $z = 50$ mm fixed, $\delta = \Delta = 2\ \mu\text{m}, 5\ \mu\text{m}, 10\ \mu\text{m}, 20\ \mu\text{m}$. How does sampling affect the performance of digital holography?

(ii) Keep $\Delta = \delta = 5\ \mu\text{m}$ fixed, while z ranges from 30 mm to 70 mm (use a small step-size). How does object distance affect the performance of digital holography?

(iii) Keep $\Delta = 5\ \mu\text{m}$ and $z = 50$ mm fixed, while δ ranges from 1 μm to 10 μm in 2 μm step size. How does discretization of the object affect the performance of digital holography?

WHERE ARE WE

Week	Date	Main Topic	Lecture	Readings	Homework Out	Homework Due
1	11-Jan-21		Introduction to computational imaging - Forward and Inverse problems - Common computational imaging problems			
	13-Jan-21		Vectors - Preliminaries			
2	18-Jan-21		**Dr. Martin Luther King, Jr. Holiday (no class)**			
	20-Jan-21	Mathematical preliminaries	Vectors and Vector Spaces - Subspaces, Finite dimensional spaces	IIP Appendix A; FSP 2.1 - 2.2		
3	25-Jan-21		Vector Spaces - Hilbert spaces	IIP Appendix B; FSP 2.3		
	27-Jan-21		Bases and Frames I - Orthonormal and Reisz Bases	IIP Appendix C; FSP 2.4 and 2.B	**HW 1**	
4	1-Feb-21		Bases and Frames II - Orthogonal Bases - Linear operators	IIP Appendix C; FSP 2.5 and 2.B		
	3-Feb-21		Fourier Analysis I - FT (1D and 2D) - FT properties	IIP 2.1, Appendix D; FSP 4.4		
5	8-Feb-21		Sampling and Interpolation - BL functions - Sampling	IIP 2.2, 2.3; FSP 5.4, 5.5		**HW 1**
	10-Feb-21		Fourier Analysis II (DFT)	IIP 2.4; FSP 3.6	**HW 2**	
6	15-Feb-21	Forward Modeling	LSI imaging: Forward problem I - Convolution	IIP 2.5 - 2.6, 3		
	17-Feb-21		LSI imaging: Forward problem I - Transfer functions	IIP 2.6		
7	22-Feb-21		LSI imaging: Forward problem I - Linear operators	IIP 3		
	24-Feb-21		LSI imaging: Forward problem I - Linear operators, Adjoints, and Inverses		**HW 3**	**HW 2**
8	1-Mar-21		**Mid-term Exams**			
	3-Mar-21		LSI imaging: Forward problem II - Sampling and Discretization: Matrix-vector form	IIP 2.7, 4		
9	8-Mar-21		LSI imaging: Forward problem II - Convolution matrix			
	10-Mar-21		LSI imaging: Forward problem II - Sampling and Discretization: Matrix-vector form - PSF, and Transfer functions			**HW3**

WE ARE HERE!

OUTLINE

- Discrete LSI Imaging Systems
- Matrix-Vector models
    - Sampling, discretization and **"transfer functions" of discrete LSI systems**

LEARNING GOALS

- Understand simple discretization schemes for object and image spaces
- Discretize image space
- Describe effect of such discretization

READING

- IIP 2.5 - 2.7
- IIP 3.4

SAMPLING
DISCRETIZATION
TRANSFER FUNCTIONS

TRANSFER FUNCTIONS - THE EIGENVALUES OF THE MATRIX REPRESENTATION

DISCRETE LINEAR SHIFT-INVARIANT SYSTEM
MATRIX-VECTOR FORMS

$$\boxed{\mathbf{A}_{mn} = \Delta_x h(x_m - x_n)}$$

- The matrix element (m, n) depends only on the difference $(x_m - x_n)$
  - Thus, we have a **discrete LSI system**
- Where, we recall that, the m**-th sample** is:

$$g_m = \sum_{n=0}^{N-1} \mathbf{A}_{mn} f_n$$

We can now specify the exact matrix-vector form!

DISCRETE LINEAR SHIFT-INVARIANT SYSTEM

MATRIX-VECTOR FORMS

Matrix-Vector form: simply rewrite the summation $g_m = \sum_{n=0}^{N-1} \mathbf{A}_{mn} f_n$, as the matrix-vector product

$$\mathbf{g} = \mathbf{A}\mathbf{f}$$

where $\mathbf{g} = \begin{bmatrix} g_0 \\ \vdots \\ g_{M-1} \end{bmatrix}$, $\quad \mathbf{f} = \begin{bmatrix} f_0 \\ \vdots \\ f_{N-1} \end{bmatrix}$,

(m, n) entry of matrix $\mathbf{A}$ is given by $\mathbf{A}_{mn} = \Delta_x h(x_m - x_n)$

IMAGING MATRIX FOR DISCRETE LSI SYSTEM
GENERAL STRUCTURE OF $\mathbf{A}$

Reminder: PSF means point spread function

PSF

$$\mathbf{g} = \underbrace{\begin{bmatrix} \mathbf{a}_0 & \mathbf{a}_{-1} & \boxed{\mathbf{a}_{-2}} & \cdots & \cdots & \mathbf{a}_{-(N-1)} \\ \mathbf{a}_1 & \mathbf{a}_0 & \mathbf{a}_{-1} & \ddots & & \vdots \\ \mathbf{a}_2 & \mathbf{a}_1 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{a}_{-1} & \mathbf{a}_{-2} \\ \vdots & & \ddots & \mathbf{a}_1 & \mathbf{a}_0 & \mathbf{a}_{-1} \\ \mathbf{a}_{M-1} & \cdots & \cdots & \mathbf{a}_2 & \mathbf{a}_1 & \mathbf{a}_0 \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} f_0 \\ f_1 \\ \\ \vdots \\ \\ f_{N-1} \end{bmatrix}$$

- The matrix $\mathbf{A}$ is Toeplitz with (m, n) entry given by $\mathbf{A}_{mn} = \mathbf{a}_{m-n}$

CIRCULANT/CYCLIC MATRICES

- The output for the discrete LSI system: $\mathbf{g} = \mathbf{A}\mathbf{f} = \mathbf{a} \star \mathbf{f}$
- Where $\star$ now denotes the **discrete convolution**

$$
\mathbf{g} = \underbrace{\left[\begin{array}{cccccc}
a_0 & a_{N-1} & a_{N-2} & \cdots & \cdots & a_{N-(M-1)} \\
a_1 & a_0 & a_{N-1} & \ddots & & \vdots \\
a_2 & a_1 & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & a_{N-1} & a_{N-2} \\
\vdots & & \ddots & a_1 & a_0 & a_{N-1} \\
a_{M-1} & \cdots & \cdots & a_2 & a_1 & a_0
\end{array}\right]}_{\mathbf{A}} \left[\begin{array}{c} f_0 \\ f_1 \\ \vdots \\ f_{M-1} \end{array}\right]
$$

PSF

DISCRETE LINEAR SHIFT-INVARIANT SYSTEM

- Have an LSI system, with **sampling set** $\{p_m(x)\}_m$, **basis set** $\{\psi_n\}_n$, if **imaging system PSF** is LSI

- Notice also that: "imaging/sensing/system matrix" $\mathbf{A}$ satisfies

$$\boxed{\mathbf{g}_m = \sum_{n=0}^{N-1} \mathbf{A}_{mn}\,\mathbf{f}_n = \sum_{n=0}^{N-1} \mathbf{a}_{m-n}\,\mathbf{f}_n}$$

1D DISCRETE CONVOLUTION

- Where $\mathbf{a} \in \mathbb{R}^M$ (a vector)

- Matrix $\mathbf{A}$ is ***Toeplitz*** with the (m, n) entry given by $\mathbf{A}_{mn} = \mathbf{a}_{m-n}$

DISCRETE LINEAR SHIFT-INVARIANT SYSTEM

DISCRETE CONVOLUTION

- The **1D discrete convolution** between the vectors $\mathbf{a} \in \mathbb{R}^M$ and $\mathbf{f} \in \mathbb{R}^M$ is:

$$\mathbf{g}_m = \sum_{n=0}^{N-1} \mathbf{a}_{m-n} \, \mathbf{f}_n$$

- Matrix $\mathbf{A}$ is ***Toeplitz*** with the (m, n) entry given by $\mathbf{A}_{mn} = \mathbf{a}_{m-n}$

DISCRETE FOURIER TRANSFORM
MATRIX REPRESENTATION

- **Discrete Fourier Transform (DFT):**

$$\hat{\mathbf{x}}_k = (F\mathbf{x})_k \qquad k \in \{0,1,\ldots,N-1\}.$$

- **Inverse discrete Fourier Transform (IDFT):**

$$\mathbf{x}_n = (F^{-1}\hat{\mathbf{x}})_n \qquad n \in \{0,1,\ldots,N-1\}.$$

- Where

$$F = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N & W_N^2 & \cdots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{(N-1)} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)^2} \end{bmatrix},$$

with $W_N^{kn} = e^{-\mathrm{j}(2\pi/N)kn}$

$$F^{-1} = \frac{1}{N}\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \cdots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \cdots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \cdots & W_N^{-(N-1)^2} \end{bmatrix} = \frac{1}{N}F^*$$

- Note that: DFT is an orthogonal basis (i.e. $F^*F = FF^* = NI$).

EIGENVALUES & EIGENVECTORS OF CIRCULANT MATRICES

$$\mathbf{g} = \mathbf{A}\mathbf{f} = \mathbf{a} \star \mathbf{f} \longleftrightarrow \hat{\mathbf{g}}_m = \hat{\mathbf{a}}_m \hat{\mathbf{f}}_m$$ (DISCRETE FT)

$$\mathbf{g} = \frac{1}{N} F^* \mathrm{diag}(\hat{\mathbf{a}}) F \mathbf{f}$$ (INVERSE DFT)

- **Discrete Transfer function** - is defined as the DFT of discrete PSF and also given by the **eigenvalues** of the circulant matrix $\mathbf{A}$
- The **eigenvectors** are the normalized DFT basis
$$\frac{1}{\sqrt{N}} (F^*)_{k'}$$
that is the k-th column of the Hermitian (or complex-transpose) of F

EIGENVALUES & EIGENVECTORS OF DISCRETE LSI SYSTEMS

- **Range:** $\mathscr{R}(\mathbf{A}) = \{\mathbf{g}_{\text{out}} : \mathbf{A}\mathbf{g}_{\text{in}} = \mathbf{g}_{\text{out}}\}$
  - The eigenvectors of $\mathbf{A}$ with non-zero eigenvalues.
- **Nullspace:** $\mathscr{N}(\mathbf{A}) = \{\mathbf{g}_{\text{in}} : \mathbf{A}\mathbf{g}_{\text{in}} = 0\}$
  - The eigenvectors of $\mathbf{A}$ with zero eigenvalues.
- **Adjoint:** $\mathbf{A}^* = \mathbf{A}^H$
  - Or, the Hermitian transpose of $\mathbf{A}$ (take complex conjugate then transpose the matrix)
- **Inverse:** $\mathbf{A}^{-1}$
  - Is the inverse of the matrix $\mathbf{A}$, such that $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$

Treat each row (or column) of object and sensor as a 1D signal & repeat for all rows (or columns).

EXTENSION TO 2D

VECTORIZATION
LEXICOGRAPHIC ORDERING

MATRIX

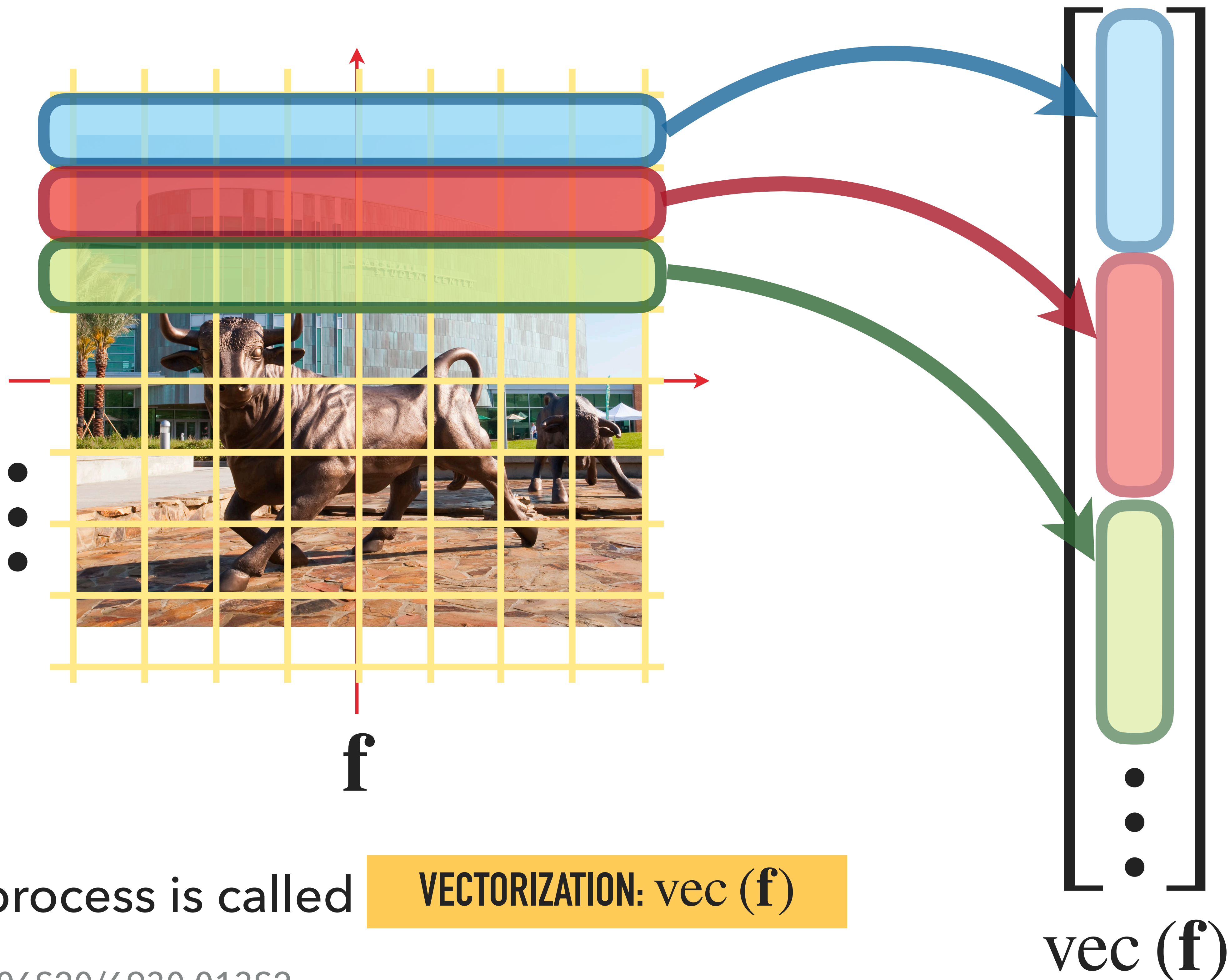
(1, 1)	(1, 2)	(1, 3)	(1, 4)	(1, 5)	(1, 6)
(2, 1)	(2, 2)	(2, 3)	(2, 4)	(2, 5)	(2, 6)
(3, 1)	(3, 2)	(3, 3)	(3, 4)	(3, 5)	(3, 6)
(4, 1)	(4, 2)	(4, 3)	(4, 4)	(4, 5)	(4, 6)
(5, 1)	(5, 2)	(5, 3)	(5, 4)	(5, 5)	(5, 6)

1	6	11	16	21	26
2	7	12	17	22	27
3	8	13	18	23	28
4	9	14	19	24	29
5	10	15	20	25	30

VECTOR

(1, 1)
(2, 1)
(3, 1)
(4, 1)
(5, 1)
(1, 2)
(2, 6)
(3, 6)
(4, 6)
(5, 6)
(1, 3)
(2, 3)
(3, 3)
(4, 3)
(5, 3)
(1, 4)
(2, 4)
(3, 4)
(4, 4)
(5, 4)
(1, 5)
(2, 5)
(3, 5)
(4, 5)
(5, 5)
(1, 6)
(2, 6)
(3, 6)
(4, 6)
(5, 6)

EXTENSION TO 2D



This process is called **VECTORIZATION: vec (f)**

2D DISCRETE LSI FORWARD MODEL

$\mathbf{g} = \mathbf{A}\mathbf{f}$, where $\mathbf{g}$ is a vectorization of

$$\mathbf{g}_{m_1,m_2} = \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} a_{m_1-n_1,m_2-n_2} f_{n_1,n_2}$$

Each sub-matrix $\mathbf{A}_{n_1}$ is a circulant matrix

$m_1 = 0$

$m_1 = 1$

$m_1 = 2$

$$\begin{bmatrix} \vdots \end{bmatrix} = \begin{pmatrix} \mathbf{A}_0 & \mathbf{A}_{N-1} & \cdots & \mathbf{A}_1 \\ \mathbf{A}_1 & \mathbf{A}_0 & \cdots & \mathbf{A}_2 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{N-1} & \mathbf{A}_{N-2} & \cdots & \mathbf{A}_0 \end{pmatrix} \begin{bmatrix} \vdots \end{bmatrix}$$

The new matrix is **block circulant**

Can you prove this extension to 2D?

2D DISCRETE LSI FORWARD MODEL

2D DISCRETE CONVOLUTION FORMULA

- This **2D LSI imaging system** in matrix vector form: $\mathbf{g} = \mathbf{A}\mathbf{f}$, where $\mathbf{g}$ is:

$$\boxed{\mathbf{g}_{m_1,m_2} = \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} a_{m_1-n_1,m_2-n_2} f_{n_1,n_2}}$$

2D DISCRETE CONVOLUTION FORMULA

- Here $\mathbf{f}$ and $\mathbf{g}$ are vectorized versions of the object and measurements, respectively.
- 2D discrete convolution gives the input-output relationship of a 2D LSI imaging system.
- The 2D discrete convolution property of the 2D DFT holds here too.

BRIEF LOOK AT 2D CASE

ANOTHER EXAMPLE

AN EXAMPLE MATRIX

- Consider the LSI system with PSF $h(x, y) = e^{-(x^2+y^2)}$,
- Using **Dirac delta sampling and Dirac object space discretization**

$$\mathbf{A}_{m_1, m_2, n_1, n_2} = \Delta^2 e^{-(x_{m_1}-x_{n_1})^2-(y_{m_2}-y_{n_2})^2}$$

  - With uniform sampling Δ: $x_{m_1} = m_1\Delta$, and $y_{m_2} = m_2\Delta$ (i.e. pixels uniformly spaced)
  - With uniform spacing δ for basis: $y_{n_1} = n_1\delta$ and $y_{n_2} = n_2\delta$
  - Thus, $\mathbf{A}_{m_1, m_2, n_1, n_2} = \Delta^2 e^{-(m_1\Delta-n_1\delta)^2-(m_2\Delta-n_2\delta)^2}$
- Thus, if $\Delta = \delta$ for example:
  - The element $\mathbf{A}_{m_1, m_2, n_1, n_2} = \Delta^2 e^{-\Delta^2(m_1-n_1)^2-\Delta^2(m_2-n_2)^2}$ depends the difference $(m_1 - n_1)$ and $(m_2 - n_2)$ not their exact values

AN EXAMPLE MATRIX

- Consider the LSI system with PSF $h(x, y) = e^{-(x^2+y^2)}$,
- **Dirac delta sampling and Dirac object space discretization (most general form):**

$$\mathbf{A}_{m_1, m_2, n_1, n_2} = \Delta^2 \, e^{-(x_{m_1}-x_{n_1})^2-(y_{m_2}-y_{n_2})^2}$$

  - Uniform object discretization and sampling
    $\mathbf{A}_{m_1, m_2, n_1, n_2} = \Delta^2 \, e^{-(m_1\Delta-n_1\delta)^2-(m_2\Delta-n_2\delta)^2}$
  - Thus, if $\Delta = \delta$ for example: $\mathbf{A}_{m_1, m_2, n_1, n_2} = \Delta^2 \, e^{-\Delta^2(m_1-n_1)^2-\Delta^2(m_2-n_2)^2}$
- **Note that here the array $\mathbf{A}$ is a 4D array (or 4D tensor)!**

DISCRETIZING LSI
IMAGING SYSTEMS

MATLAB PRACTICE 8

DISCRETE LSI IMAGING SYSTEMS

- Discrete convolution (1D and 2D)
  - DFT diagonalizes Circulant (i.e., Toeplitz or Hankel) matrices
- Discretize 1D and 2D LSI systems

WHAT WE COVERED TODAY

- Discretizing LSI imaging systems
  - Imaging systems represented as a linear system of equations (Matrix-Vector equations)
    - Sampling (imaging sensor/detector)
    - Discretization of object space and imaging system
    - Discrete transfer functions

SEE YOU NEXT TIME!