
Lab 2C: You Can't Go Home Again

— Weichien Liao —
2018/10/08

Review and Prerequisite

The Blur Matrix

$$B_m = \begin{bmatrix} \frac{1}{2} & \frac{1}{4} & & & \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & & \\ & \ddots & \ddots & \ddots & \\ & & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ & & & \frac{1}{4} & \frac{1}{2} \end{bmatrix}_{m \times m}$$

Blurring an Image

- Let X be an $m \times n$ pixel intensity matrix of an image. Then the following equation applies local blurring k times to X in each direction:

$$Z = VXH$$

where $V = (B_m)^k$ and $H = (B_n)^k$.

Solving Linear System in MATLAB

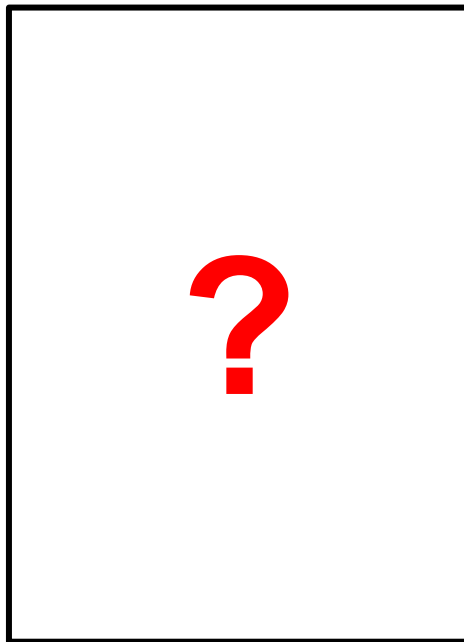
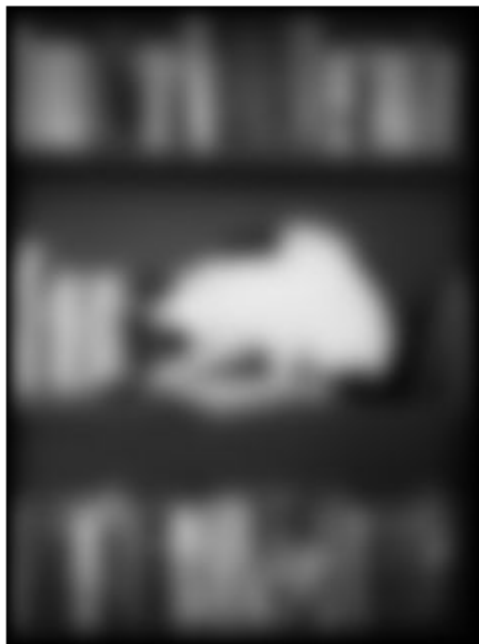
- Solving $Ax = b$ using Gaussian Elimination can be done using the following line:

$$x = A \setminus b$$

- What if the linear system has multiple right hand sides, i.e. the right hand side is an m -by- n matrix instead of a column vector?
- What if we want to solve $y^T A = b^T$.

Hands on: Lab 2C You Can't Go Home Again

Deblur: Example



Deblur: Restoring a Blurred Image

- The process of restoring a blurred image can be expressed using matrix inverses:

$$X = V^{-k} Z H^{-k}$$

- In computational practice, the above equation is computed by solving linear systems.

Problems

1. Write a function which computes the inverse of the given blur matrix.

([Link](#) for detailed instructions)

2. Write a function that returns the restored image for a given blurred image.

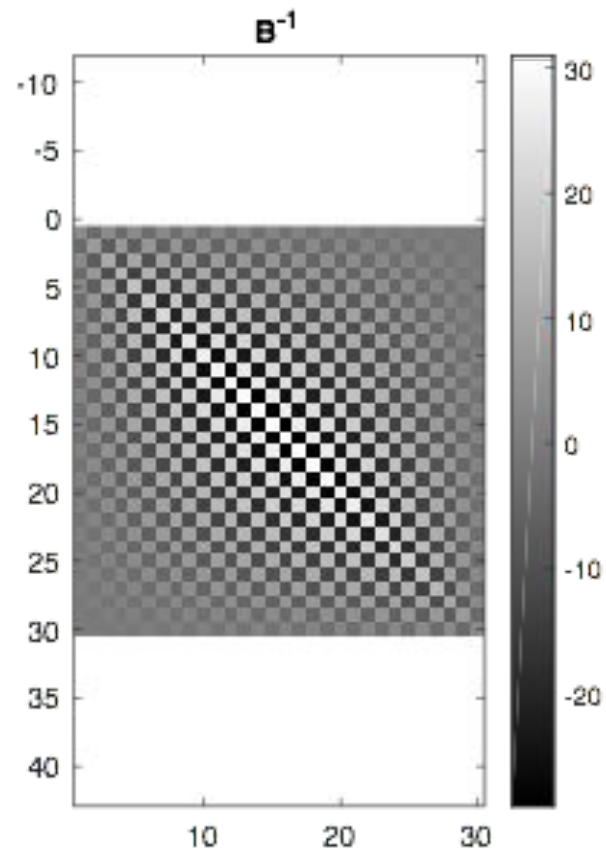
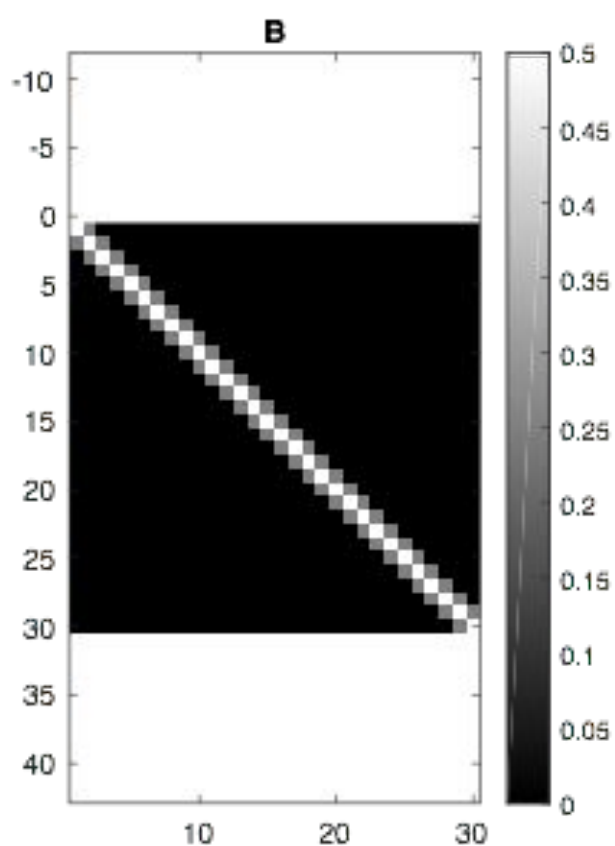
([Link](#) for detailed instructions)

What We Learned

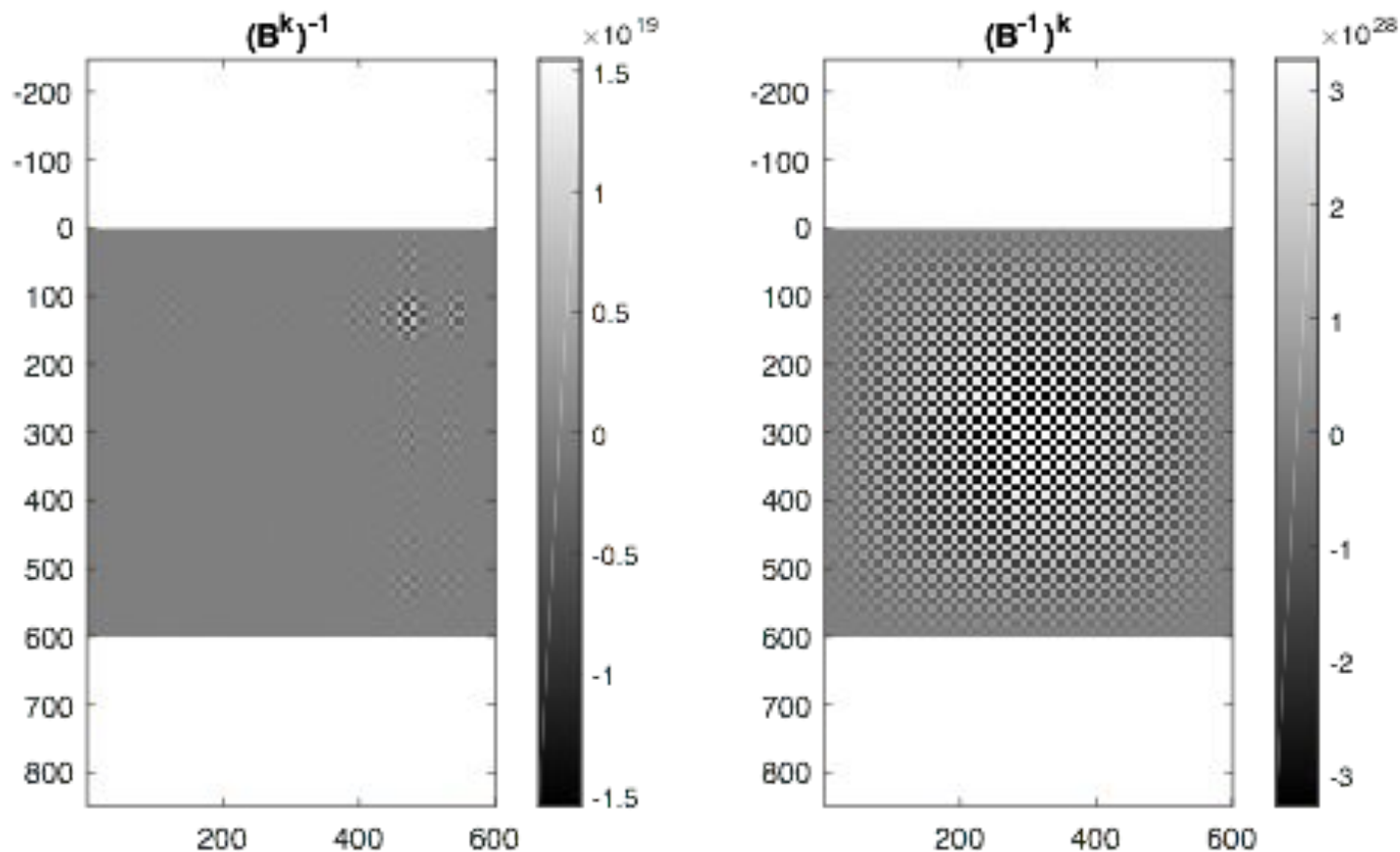
Solving Linear Systems in MATLAB

- Solve $Ax = b$ in MATLAB by the command `\`
- Compute the inverse of a matrix by solving linear systems
- Multiply by an inverse on the left and multiply by an inverse on the right

B vs $\text{inv}(B)$



Order of Inverse and Power Matters



Blurring and Deblurring Images

- An image can be blurred as much as we like.
- A severely blurred image cannot be accurately restored in finite precision.