Mathematical Beer Goggles or The Mathematics of Image Processing

Melina Freitag

Department of Mathematical Sciences
University of Bath

Postgraduate Seminar Series
University of Bath
12th February 2008

Outline

Outline

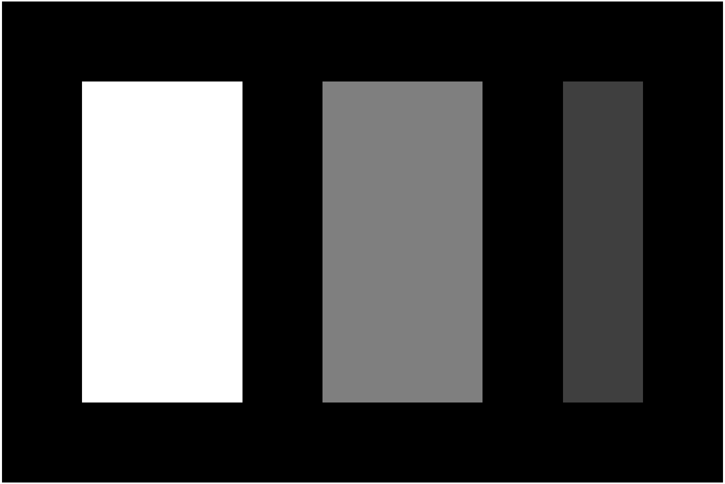
Grayscale images

One dimensional matrix

$$X = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 8 & 8 & 0 & 4 & 4 & 0 & 2 & 0 \\ 0 & 8 & 8 & 0 & 4 & 4 & 0 & 2 & 0 \\ 0 & 8 & 8 & 0 & 4 & 4 & 0 & 2 & 0 \\ 0 & 8 & 8 & 0 & 4 & 4 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

```
imagesc(X), colormap(gray)
```

MATLAB image



RGB images

Three dimensional matrix

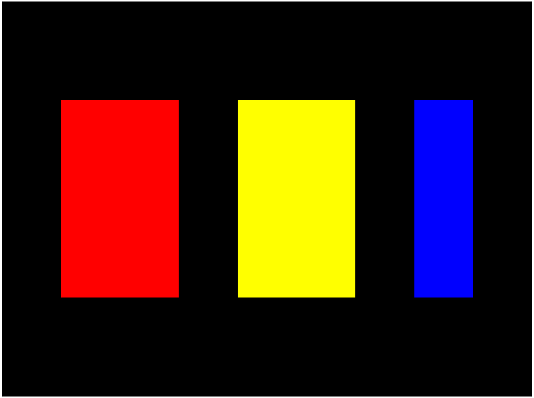
$$X(:,:,1) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$X(:,:,2) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$X(:,:,3) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

```
imagesc(X)
```

MATLAB image



```
X = imread('pic.jpg'), imwrite(X,'pic.jpg')
```

Outline

The singular value decomposition (SVD)

Existence and Uniqueness

Let $X \in \mathbb{C}^{m,n}$, $m \geq n$ Then

$$\begin{bmatrix} & X & \end{bmatrix} \left[v_1|v_2|\ldots|v_n\right] = \left[u_1|u_2|\ldots|u_m\right] \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \\ & & & 0 \end{bmatrix}$$

or

$$X = U\Sigma V^T,$$

where $U^TU = I$, with columns of U called left singular vectors and $V^TV = I$ with right singular vectors as columns of V and $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_n)$ called singular values ordered such that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$.

Low-rank approximations

Theorem (The rank of a matrix)

The rank of X is r, the number of nonzero singular values in

$$X = U\Sigma V^T = U \begin{bmatrix} \sigma_1 & & & & & \\ & \sigma_2 & & & & \\ & & \ddots & & & \\ & & & \sigma_r & & \\ & & & & 0 & \cdots \\ & & & & 0 & \ddots \end{bmatrix} V^T.$$

Low-rank approximations

Theorem (The rank of a matrix)

The rank of X is r, the number of nonzero singular values in

$$X = U\Sigma V^T = U \begin{bmatrix} \sigma_1 & & & & & \\ & \sigma_2 & & & & \\ & & \ddots & & & \\ & & & \sigma_r & & \\ & & & & 0 & \cdots \\ & & & & 0 & \ddots \end{bmatrix} V^T.$$

Theorem (Another representation)

X is the sum of r rank-one matrices

$$X = \sum_{j=1}^{r} \sigma_j u_j v_j^T.$$

Low-rank approximations

Theorem

For any ν with $0 \leq \nu \leq r$, define

$$X_\nu = \sum_{j=1}^{\nu} \sigma_j u_j v_j^T,$$

Then

$$\|X - X_\nu\|_2 = \inf_{B \in \mathbb{C}^{m,n},\, rank(B) \leq \nu} \|X - B\|_2 = \sigma_{\nu+1}.$$

Low-rank approximations

Proof ($m = n$)

$$\begin{aligned} \|X - X_\nu\|_2 &= \|\sum_{j=\nu+1}^{r} \sigma_j u_j v_j^T\|_2 = \|U \begin{bmatrix} 0 & & & \\ & \sigma_{\nu+1} & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} V^T\|_2 \\ &= \sigma_{\nu+1} \end{aligned}$$

Remains to show that there is no closer rank ν matrix to X.

Low-rank approximations

Proof ($m = n$)

$$\begin{aligned} \|X - X_\nu\|_2 &= \|\sum_{j=\nu+1}^{r} \sigma_j u_j v_j^T\|_2 = \|U \begin{bmatrix} 0 & & & \\ & \sigma_{\nu+1} & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} V^T\|_2 \\ &= \sigma_{\nu+1} \end{aligned}$$

Remains to show that there is no closer rank ν matrix to X.

- Let B have rank ν, null space of (B) has dimension $n - \nu$
- $\{v_1, \ldots, v_{\nu+1}\}$ has dimension $\nu + 1$
- Let h be a unit vector in their intersection:

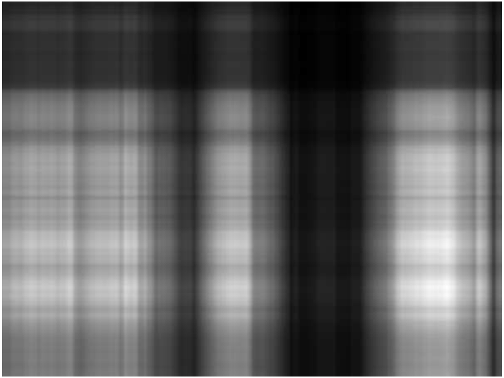
$$\begin{aligned} \|X - B\|^2 &\geq \|(X - B)h\|^2 = \|Xh\|^2 = \|U\Sigma V^T h\|^2 \\ &= \|\Sigma(V^T h)\|^2 \geq \sigma_{\nu+1}^2 \|V^T h\|^2 \geq \sigma_{\nu+1}^2. \end{aligned}$$

Example $m = 604$, $n = 453$, $m * n = 273612$

Compression ratio

$$c := \frac{(m+n)\nu}{mn}$$

Rank-1 approximation $m = 604$, $n = 453$, $m + n = 1057$



Compression ratio

$$c := 3.8631e-03$$

Rank-2 approximation

Compression ratio

$$c := 7.7263e - 03$$

Rank-3 approximation

Compression ratio

$$c := 1.1589e-02$$

Rank-4 approximation

Compression ratio

$$c := 1.5453e-02$$

Rank-5 approximation

Compression ratio

$$c := 1.9316e - 02$$

Rank-10 approximation

Compression ratio

$$c := 3.8631e - 02$$

Rank-20 approximation

Compression ratio

$$c := 7.7263e - 02$$

Rank-30 approximation

Compression ratio

$$c := 0.11589$$

Rank-40 approximation

Compression ratio

$$c := 0.15453$$

Rank-60 approximation

Compression ratio

$$c := 0.23179$$

Rank-80 approximation

Compression ratio

$$c := 0.30905$$

Outline

Blurred and exact images

- Let X be the exact image
- Let B be the blurred image

Blurred and exact images

- Let X be the exact image
- Let B be the blurred image
- If the blurring of the columns is independent of the blurring in the rows then

$$A_c X A_r^T = B, \quad A_c \in \mathbb{R}^{m,m}, \quad A_r \in \mathbb{R}^{n,n}$$

Blurred and exact images

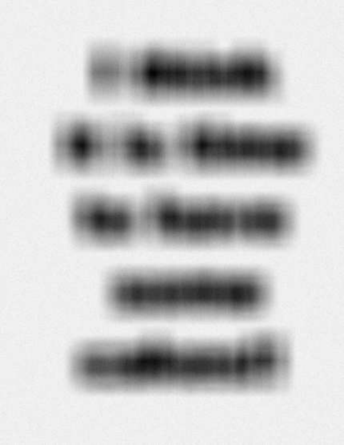
- Let X be the exact image
- Let B be the blurred image
- If the blurring of the columns is independent of the blurring in the rows then

$$A_c X A_r^T = B, \quad A_c \in \mathbb{R}^{m,m}, \quad A_r \in \mathbb{R}^{n,n}$$

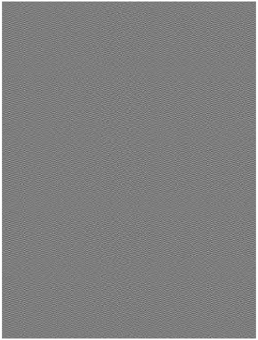
First attempt at deblurring

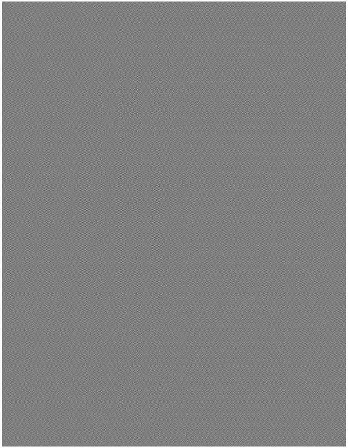
$$X_{\text{Naive}} = A_c^{-1} B A_r^{-T}.$$

First attempt at deblurring



$X_{\text{Naive}} = A_c^{-1} B A_r^{-T}$





What is the problem?

A noisy blurred image

$$B = B_{\text{exact}} + E = A_c X A_r^T + E$$

and therefore

$$X_{\text{Naive}} = X + A_c^{-1} E A_r^{-T}.$$

What is the problem?

A noisy blurred image

$$B = B_{\text{exact}} + E = A_c X A_r^T + E$$

and therefore

$$X_{\text{Naive}} = X + A_c^{-1} E A_r^{-T}.$$

Error

The naive solution satisfies

$$\frac{\|X_{\text{Naive}} - X\|_F}{\|X\|_F} \leq \text{cond}(A_c)\text{cond}(A_r)\frac{\|E\|_F}{\|B\|_F}.$$

Deblurring using a general model

The blurring process as a linear model

We assume the blurring process is linear, i.e.

$$x = \mathrm{vec}(X) = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \in \mathbb{R}^N, \quad b = \mathrm{vec}(B) = \begin{bmatrix} b_1 \\ \vdots \\ b_N \end{bmatrix} \in \mathbb{R}^N$$

$N = m * n$ are related by the linear model

$$Ax = b$$

Deblurring using a general model

The blurring process as a linear model

We assume the blurring process is linear, i.e.

$$x = \text{vec}(X) = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \in \mathbb{R}^N, \quad b = \text{vec}(B) = \begin{bmatrix} b_1 \\ \vdots \\ b_N \end{bmatrix} \in \mathbb{R}^N$$

$N = m * n$ are related by the linear model

$$Ax = b$$

$b = b_{exact} + e$

$$x_{\text{Naive}} = A^{-1}b = A^{-1}b_{\text{exact}} + A^{-1}e = x + A^{-1}e$$

Separable two dimensional blurs

The Kronecker product

If horizontal and vertical flow can be separated then

$$Ax = b \quad \Leftrightarrow \quad A\mathrm{vec}(X) = \mathrm{vec}(B) = \mathrm{vec}(A_c X A_r^T)$$

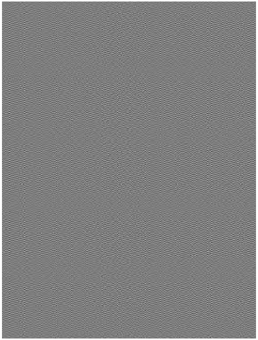
$$(A_r \otimes A_c)\mathrm{vec}(X) = \mathrm{vec}(A_c X A_r^T),$$

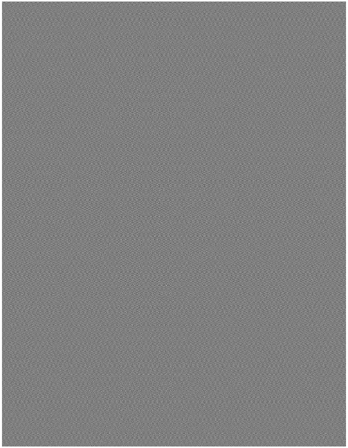
where

$$A = A_r \otimes A_c = \begin{bmatrix} a_{11}^r A_c & \dots & a_{1n}^r A_c \\ \vdots & \vdots & \vdots \\ a_{n1}^r A_c & \dots & a_{nn}^r A_c \end{bmatrix},$$

$$(U_r \Sigma_r V_r^T) \otimes (U_c \Sigma_c V_c^T) = (U_r \otimes U_c)(\Sigma_r \otimes \Sigma_c)(V_r \otimes V_c)^T.$$

$x_{\text{Naive}} = x + A^{-1}e$





Outline

Taking bad pictures

Sources of bad pictures

- defocus the camera lens (limitations in the optical system)
- motion blur
- air turbulence
- atmospheric blurring

Taking bad pictures

Sources of bad pictures

- defocus the camera lens (limitations in the optical system)
- motion blur
- air turbulence
- atmospheric blurring

Noise E

- background photons from both natural or artificial sources
- signal represented by finite number of bits (quantisation error)

Modelling the blurring matrix A

Single bright pixel

$$x = e_i \Rightarrow Ae_i = \text{ column } i \text{ of } A$$

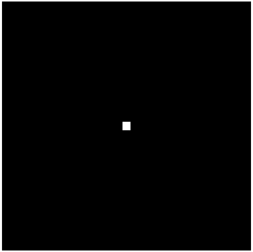


Figure: Point source (single bright pixel)

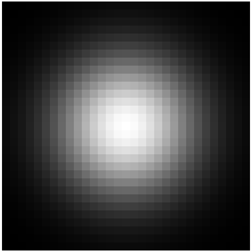


Figure: Point spread function (PSF)

Modelling the blurring matrix A

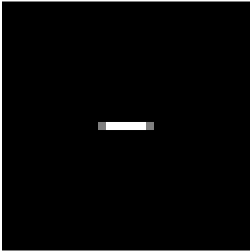


Figure: Motion blur

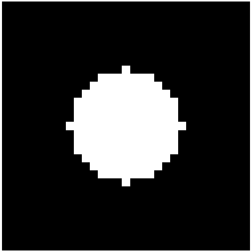


Figure: Out-of-focus blur

$$p_{ij} = \begin{cases} 1/(\pi r)^2 & \text{if} \quad (i-k)^2 + (j-l)^2 \leq r^2 \\ 0 & \text{otherwise} \end{cases}$$

Modelling the blurring matrix A

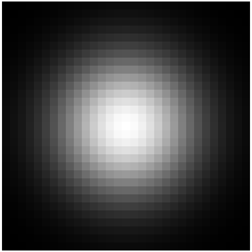


Figure: Atmospheric turbulence blur

$$p_{ij} = \exp\left(-\frac{1}{2}\begin{bmatrix} i-k \\ j-l \end{bmatrix}^T \begin{bmatrix} s_1^2 & \rho^2 \\ \rho^2 & s_2^2 \end{bmatrix}^{-1} \begin{bmatrix} i-k \\ j-l \end{bmatrix}^T\right)$$

$$p_{ij} = \left(1 + \begin{bmatrix} i-k \\ j-l \end{bmatrix}^T \begin{bmatrix} s_1^2 & \rho^2 \\ \rho^2 & s_2^2 \end{bmatrix}^{-1} \begin{bmatrix} i-k \\ j-l \end{bmatrix}^T\right)^{-\beta}$$

Boundary conditions and structured matrix computations

Boundary conditions

- Zero boundary conditions
- Periodic boundary conditions
- Reflexive boundary conditions

The matrix A which is obtained from P by convolution becomes

- Block Toeplitz matrix
- Block Circulant matrix
- Sum of Block Toeplitz and Block Hankel and Block Toeplitz plus Hankel matrices

Spectral filtering

The SVD again

With

$$A = U\Sigma V^T = \left[u_1 | u_2 | \dots | u_N\right] \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_N \end{bmatrix} \begin{bmatrix} v_1^T \\ \vdots \\ v_N^T \end{bmatrix}$$

we have

$$x_{\text{Naive}} = A^{-1}b = V\Sigma^{-1}U^T b = \sum_{i=1}^{N} \frac{u_i^T b}{\sigma_i} v_i$$

$$X_{\text{Naive}} = \sum_{i=1}^{N} \frac{u_i^T b}{\sigma_i} V_i = \sum_{i=1}^{N} \frac{u_i^T b_{\text{exact}}}{\sigma_i} V_i + \sum_{i=1}^{N} \frac{u_i^T e}{\sigma_i} V_i$$

Spectral filtering

Behaviour of singular values

- $\sigma_i \to 0$ as i grows
- the more "blurry" the function, the faster the decay rate
- $\mathrm{cond}(A) = \sigma_1 / \sigma_N$

Spectral filtering

Behaviour of singular values

- $\sigma_i \to 0$ as i grows
- the more "blurry" the function, the faster the decay rate
- $\mathrm{cond}(A) = \sigma_1/\sigma_N$

The regularised solution

Introduce filter factors Φ_i

$$x_{\text{Naive}} = \sum_{i=1}^{N} \Phi_i \frac{u_i^T b}{\sigma_i} v_i$$

Two methods

TSVD

$$\Phi_i = \begin{cases} 1 & i = 1, \ldots, k \\ 0 & i = k+1, \ldots, N \end{cases}$$

Two methods

TSVD

$$\Phi_i = \begin{cases} 1 & i = 1, \ldots, k \\ 0 & i = k+1, \ldots, N \end{cases}$$

Tikhonov regularisation

$$\Phi_i = \frac{\sigma_i^2}{\sigma_i^2 + \alpha^2}$$

where $\alpha > 0$ is a regularisation parameter, This choice of filter factor yields solution to the minimisation problem

$$\min_x \{\|b - Ax\|_2^2 + \alpha^2 \|x\|_2^2\}.$$

Regularisation and perturbation errors

Regularised solution

$$\begin{aligned} x_{\text{filt}} &= V\Phi\Sigma^{-1}U^T b \\ &= V\Phi\Sigma^{-1}U^T A x_{\text{exact}} + V\Phi\Sigma^{-1}U^T e \\ &= V\Phi V^T x_{\text{exact}} + V\Phi\Sigma^{-1}U^T e \end{aligned}$$

$$x_{\text{exact}} - x_{\text{filt}} = \underbrace{(I - V\Phi V^T)x_{\text{exact}}}_{\text{Regularisation error}} - \underbrace{V\Phi\Sigma^{-1}U^T e}_{\text{Perturbation error}}$$

Regularisation and perturbation errors

Regularised solution

$$
\begin{aligned}
x_{\text{filt}} &= V\Phi\Sigma^{-1}U^T b \\
&= V\Phi\Sigma^{-1}U^T A x_{\text{exact}} + V\Phi\Sigma^{-1}U^T e \\
&= V\Phi V^T x_{\text{exact}} + V\Phi\Sigma^{-1}U^T e
\end{aligned}
$$

$$
x_{\text{exact}} - x_{\text{filt}} = \underbrace{(I - V\Phi V^T)x_{\text{exact}}}_{\text{Regularisation error}} - \underbrace{V\Phi\Sigma^{-1}U^T e}_{\text{Perturbation error}}
$$

Oversmoothing and undersmoothing

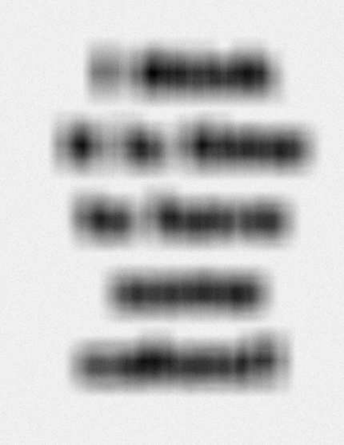
- small regularisation error, large perturbation error leads to undersmoothed solution
- large regularisation error, small perturbation error leads to oversmoothed solution

Regularisation and perturbation errors

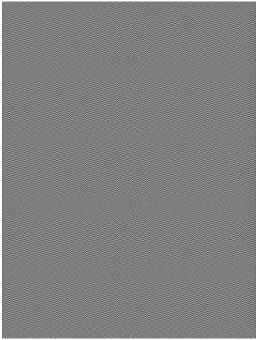
Parameter choice methods

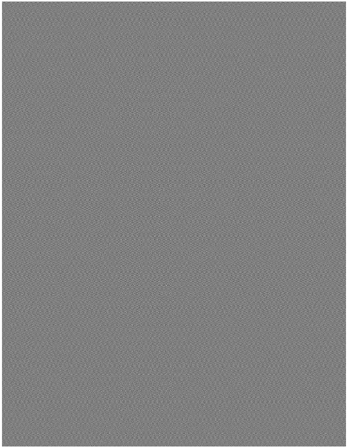
- Discrepancy Principle
- Generalised Cross Validation
- L-Curve Criterion

A second attempt at deblurring



$X_{\text{Naive}} = A_c^{-1} B A_r^{-T}$





Filtered solution using TSVD

Figure: $k = 4801$, $N = 83000$

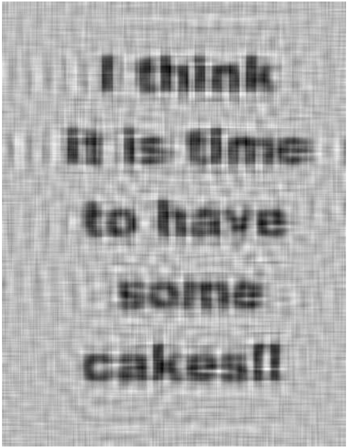


Figure: $k = 6630$, $N = 238650s$

Tikhonov regularisation

Figure: $\alpha = 0.0276$

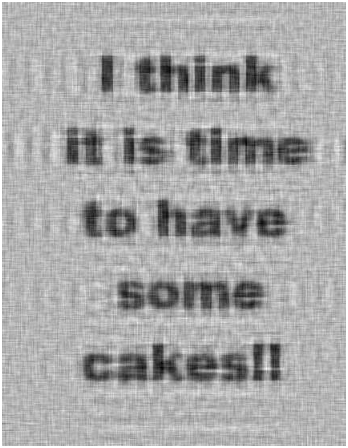


Figure: $\alpha = 0.0137$

P. C. Hansen, J. G. Nagy, and D. P. O'Leary, *Deblurring Images - Matrices, Spectra and Filtering*, SIAM, Philadelphia, 1st ed., 2006.