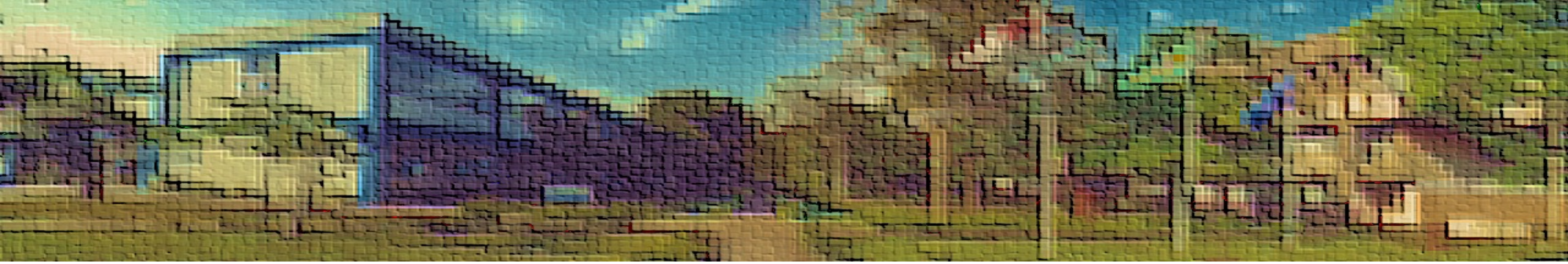


MAC317 Introdução ao Processamento de Sinais Digitais

Prof. Marcel P. Jackowski

mjack@ime.usp.br

Aula #14: Convolução e filtragem

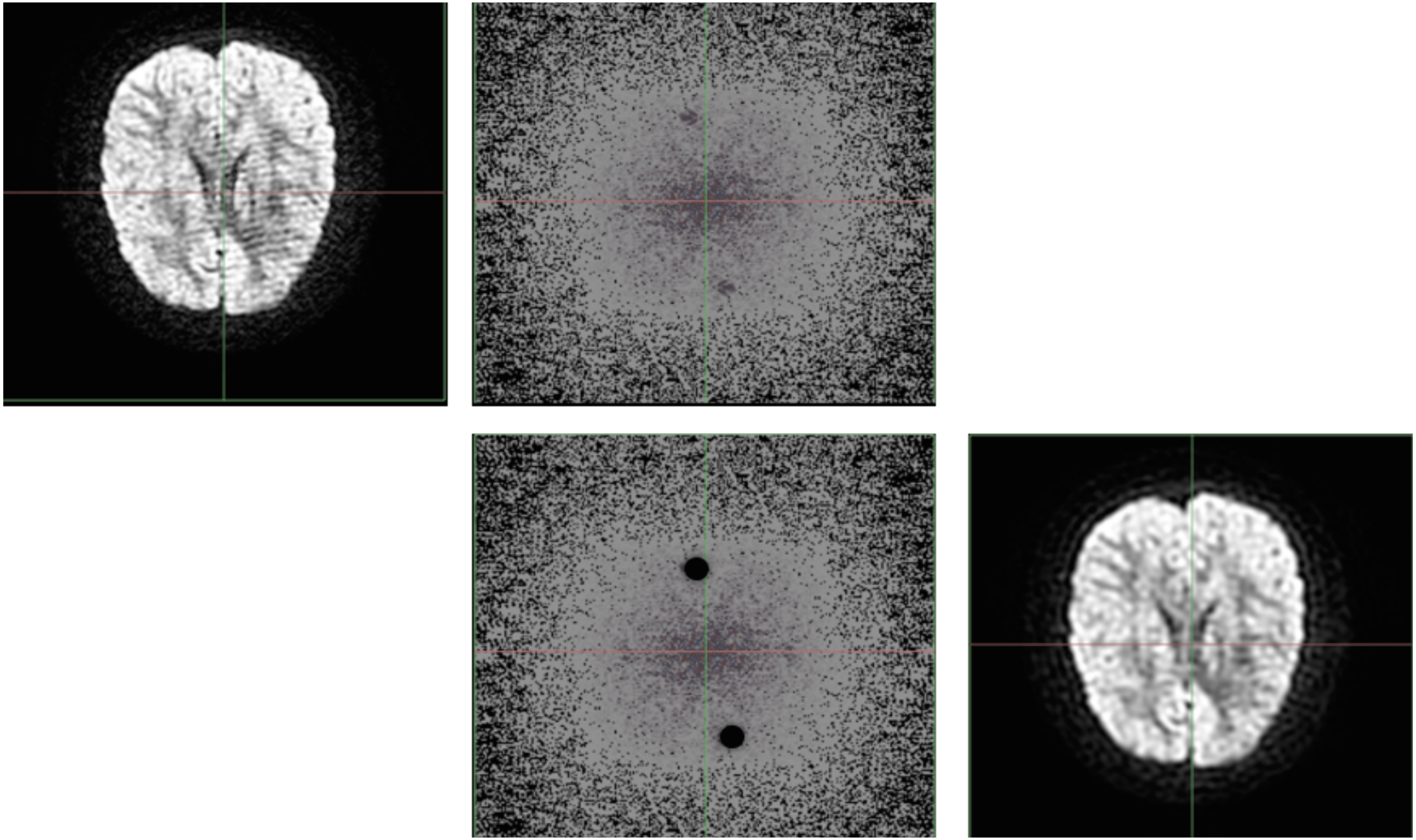
Introdução

- O termo *filtragem* refere-se à alteração sistemática de conteúdo de frequência(s) de um sinal ou imagem
  - Em particular, desejamos filtrar algumas frequências
- Esta operação é de natureza linear e normalmente realizada no domínio do tempo ou frequência
- A *convolução* é uma ferramenta importante para filtragem no domínio do tempo
  - A forma da convolução depende do espaço vetorial no qual os sinais residem

Remoção de ruído

- A sequência típica de operações para a remoção de ruído de um sinal no domínio de frquências:
  - Transformar o sinal para o domínio de frequências utilizando DFT
  - Zerar o(s) componente(s) correspondentes às frequências desejadas
  - Reconstruir o sinal utilizando a transformada DFT inversa

Remoção de ruído



Remoção de altas frequências

- Utilizando a média das amostras

A simple approach to partially removing the high frequencies from the signal while leaving low frequencies is to use a *moving average*, by replacing each sample x_k with the average value of itself and nearby samples. For example, let us take $\mathbf{w} \in \mathbb{R}^N$ as the vector defined by

$$w_k = \frac{1}{2}x_{k-1} + \frac{1}{2}x_k \tag{4.1}$$

for $0 \leq k \leq N-1$. Equation (4.1) presents a problem in the case where $k = 0$, since x_{-1} is undefined. We will thus interpret all indexes "modulo N," so, for example, $x_{-1} = x_{N-1}$. This is in keeping with the assumption that the function $x(t)$ is periodic with period 1, so $x_{-1} = x(-1/N) = x((N-1)/N) = x_{N-1}$. We can thus consider x_k as defined for all k via $x_k = x_{k \bmod N}$, if necessary.

Comportamento da média

$x(t) = \sin(2\pi q t)$, then

$$
\begin{aligned}
w_k &= \frac{1}{2}x_{k-1} + \frac{1}{2}x_k \\
&= \frac{1}{2}\sin\left(\frac{2\pi q(k-1)}{N}\right) + \frac{1}{2}\sin\left(\frac{2\pi qk}{N}\right) \\
&= \left(\frac{1+\cos(2\pi q/N)}{2}\right)\sin\left(\frac{2\pi qk}{N}\right) - \frac{1}{2}\sin\left(\frac{2\pi q}{N}\right)\cos\left(\frac{2\pi qk}{N}\right) \\
&= A\sin\left(\frac{2\pi qk}{N}\right) - B\cos\left(\frac{2\pi qk}{N}\right),
\end{aligned}
$$

where $A = (1+\cos(2\pi q/N))/2$, $B = \frac{1}{2}\sin(2\pi q/N)$, and we make use of $\sin(a-b) = \sin(a)\cos(b) - \cos(a)\sin(b)$ with $a = 2\pi qk/N$, $b = 2\pi q/N$. If q is close to zero (more accurately, if q/N is close to zero), then $A \approx 1$ and $B \approx 0$. As a consequence, $w_k \approx \sin(2\pi qk/N) = x_k$. In short, a low-frequency waveform passes through the two-point averaging process largely unchanged. On the other hand, if $q \approx N/2$ (the Nyquist frequency), then $A \approx 0$ and $B \approx 0$, so $w_k \approx 0$. The highest frequencies that we can represent will be nearly zeroed out by this process.

Resultado da filtragem

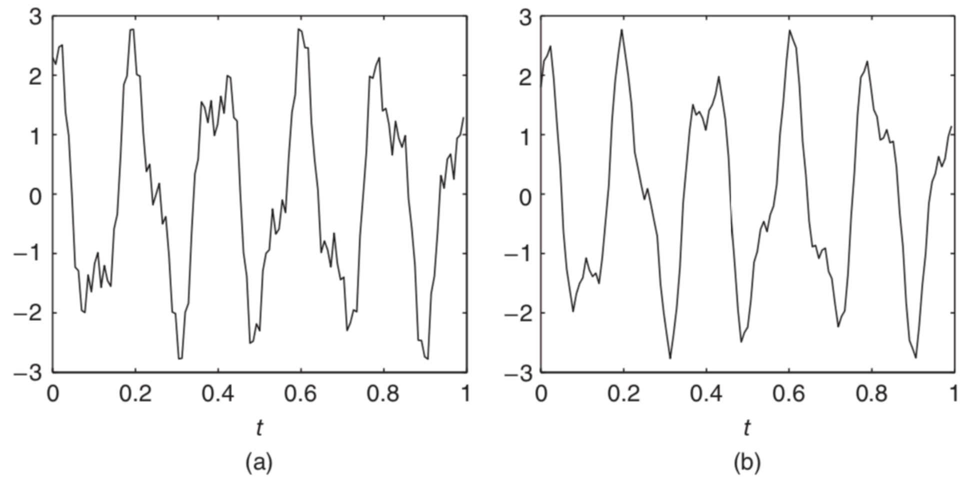
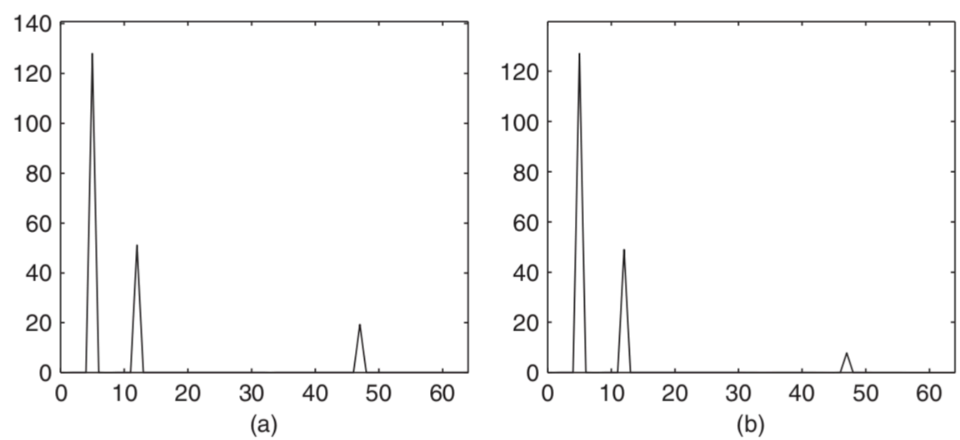


Figure 4.1 Original and smoothed signal.



Convolução

The low-pass filtering operation above is a special case of convolution, an operation that plays an important role in signal and image processing, and indeed many areas of mathematics. In what follows, we will assume that all vectors in $\mathbb{C}^N$ are indexed from 0 to $N-1$. Moreover, when convenient, we will assume that the vectors have been extended periodically with period N in the relevant index, via $x_k = x_{k \bmod N}$.

Remark 4.1 If we extend a vector $\mathbf{x}$ periodically to all index values k via $x_k = x_{k \bmod N}$, then for any value of m we have

$$\sum_{k=m}^{m+N-1} x_k = \sum_{k=0}^{N-1} x_{k+m} = \sum_{k=0}^{N-1} x_k.$$

Definição de convolução

Let us recast the filtering operation above in a more general format. We begin with a definition.

Definition 4.1 Let $\mathbf{x}$ and $\mathbf{y}$ be vectors in $\mathbb{C}^N$. The circular convolution of $\mathbf{x}$ and $\mathbf{y}$ is the vector $\mathbf{w} \in \mathbb{C}^N$ with components

$$w_r = \sum_{k=0}^{N-1} x_k y_{(r-k) \bmod N} \tag{4.2}$$

for $0 \leq r \leq N - 1$. The circular convolution is denoted $\mathbf{w} = \mathbf{x} * \mathbf{y}$.

Convolução

We compute the quantity w_0 by taking the vector $\mathbf{x}$ and the vector $\mathbf{y}$ indexed in reverse order starting at $k = 0$, and lining them up:

$$\begin{array}{ccccc} x_0 & x_1 & x_2 & \cdots & x_{N-1} \\ y_0 & y_{N-1} & y_{N-2} & \cdots & y_1. \end{array}$$

Then w_0 is simply the dot product of these two rows, $w_0 = x_0y_0 + x_1y_{N-1} + \cdots + x_{N-1}y_1$. To compute w_1, take the y_1 at the end of the second row and move it to the front, pushing other components to the right, as

$$\begin{array}{ccccc} x_0 & x_1 & x_2 & \cdots & x_{N-1} \\ y_1 & y_0 & y_{N-1} & \cdots & y_2. \end{array}$$

Finalmente

The overall computation can be summarized as

$$
\begin{array}{c|ccccc}
 & x_0 & x_1 & x_2 & \cdots & x_{N-1} \\
\hline
w_0 & y_0 & y_{N-1} & y_{N-2} & \cdots & y_1 \\
w_1 & y_1 & y_0 & y_{N-1} & \cdots & y_2 \\
w_2 & y_2 & y_1 & y_0 & \cdots & y_3 \\
 & & & & \vdots & \\
w_{N-1} & y_{N-1} & y_{N-2} & y_{N-3} & \cdots & y_0.
\end{array}
$$

Each w_r is obtained as the dot product of the corresponding row with the vector $(x_0, x_1, \ldots, x_{N-1})$.

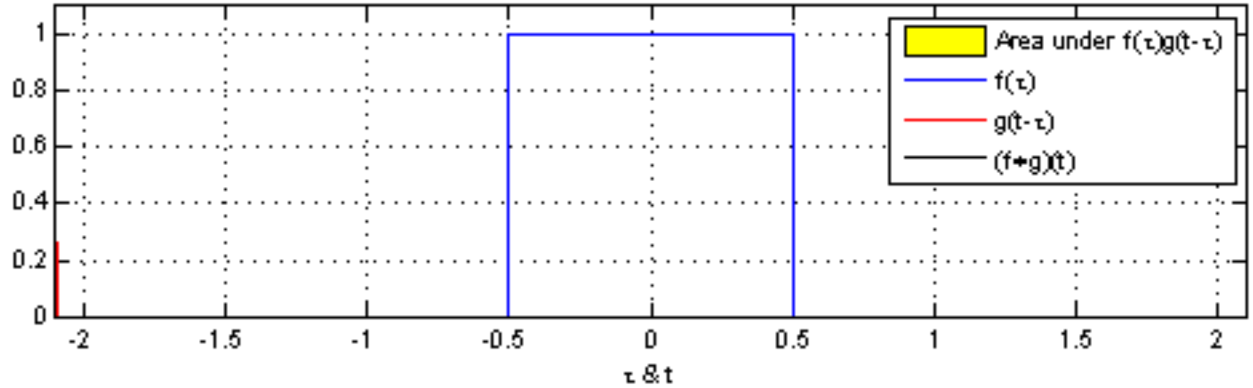
Convolução

The expression

$$s_o(x) = \int_{-\infty}^{+\infty} s_i(\xi)h(x-\xi)d\xi = s_i * h$$

is called *convolution*, defined as:

$$s_1(x) * s_2(x) = \int_{-\infty}^{+\infty} s_1(\xi)s_2(x-\xi)d\xi = \int_{-\infty}^{+\infty} s_1(x-\xi)s_2(\xi)d\xi$$



Exemplo

- O filtro passa-baixa da média entre dois pontos que acabamos de ver pode ser reescrito como uma convolução

Example 4.2 *The two-point low-pass filtering operation from Section 4.2.1 can be cast as a convolution* $\mathbf{w} = \mathbf{x} * \ell$ *where* $\ell \in \mathbb{C}^{128}$ *has components* $\ell_0 = \frac{1}{2}$, $\ell_1 = \frac{1}{2}$, *and all other* $\ell_k = 0$. *To see this note that* $\ell_{(r-k) \bmod N} = \frac{1}{2}$ *when* $k = r$ *or* $k = r - 1$ *and* 0 *otherwise. Equation (4.2) then becomes* $w_r = x_r/2 + x_{r-1}/2$, *which is equation (4.1). Indeed all the filtering operations with which we will be concerned can be implemented via an appropriate convolution.*

Propriedades

Theorem 4.1 *Let* $\mathbf{x}$, $\mathbf{y}$, *and* $\mathbf{w}$ *be vectors in* $\mathbb{C}^N$. *The following hold:*

1. *Linearity:* $\mathbf{x} * (a\mathbf{y} + b\mathbf{w}) = a(\mathbf{x} * \mathbf{y}) + b(\mathbf{x} * \mathbf{w})$ *for any scalars* a, b.
2. *Commutativity:* $\mathbf{x} * \mathbf{y} = \mathbf{y} * \mathbf{x}$.
3. *Matrix formulation: If* $\mathbf{w} = \mathbf{x} * \mathbf{y}$, *then* $\mathbf{w} = \mathbf{M}_\mathbf{y}\mathbf{x}$, *where* $\mathbf{M}_\mathbf{y}$ *is the* $N \times N$ *matrix*

$$\mathbf{M}_\mathbf{y} = \begin{bmatrix} y_0 & y_{N-1} & y_{N-2} & \cdots & y_1 \\ y_1 & y_0 & y_{N-1} & \cdots & y_2 \\ y_2 & y_1 & y_0 & \cdots & y_3 \\ & & & \vdots & \\ y_{N-1} & y_{N-2} & y_{N-3} & \cdots & y_0 \end{bmatrix}.$$

   In particular, the row k *and column* m *entries of* $\mathbf{M}_\mathbf{y}$ *are* $y_{(k-m) \bmod N}$, *rows and columns indexed from 0. Moreover,* $\mathbf{M}_\mathbf{x}\mathbf{M}_\mathbf{y} = \mathbf{M}_{\mathbf{x}*\mathbf{y}}$.
   The matrix $\mathbf{M}_\mathbf{y}$ *is called the* circulant matrix *for* $\mathbf{y}$. *Note that the rows of* $\mathbf{M}_y$, *or the columns, can be obtained by the circular shifting procedure described after equation (4.2).*
4. *Associativity:* $\mathbf{x} * (\mathbf{y} * \mathbf{w}) = (\mathbf{x} * \mathbf{y}) * \mathbf{w}$.
5. *Periodicity: If* x_k *and* y_k *are extended to be defined for all* k *with period* N, *then the quantity* w_r *defined by equation (4.2) is defined for all* r *and satisfies* $w_r = w_{r \bmod N}$.

Teorema da convolução

Computing the convolution of two vectors in the time domain may look a bit complicated, but the frequency domain manifestation of convolution is very simple.

Theorem 4.2 ***The Convolution Theorem*** *Let* $\mathbf{x}$ *and* $\mathbf{y}$ *be vectors in* $\mathbb{C}^N$ *with DFT's* $\mathbf{X}$ *and* $\mathbf{Y}$*, respectively. Let* $\mathbf{w} = \mathbf{x} * \mathbf{y}$ *have DFT* $\mathbf{W}$*. Then*

$$W_k = X_k Y_k \tag{4.4}$$

for $0 \leq k \leq N - 1$.

Teorema da convolução

$$W_k = \sum_{m=0}^{N-1} e^{-2\pi ikm/N} w_m.$$

Since $\mathbf{w} = \mathbf{x} * \mathbf{y}$, we have $w_m = \sum_{r=0}^{N-1} x_r y_{m-r}$. Substitute this into the formula for W_k above and interchange the summation order to find

$$W_k = \sum_{r=0}^{N-1} \sum_{m=0}^{N-1} e^{-2\pi ikm/N} x_r y_{m-r}.$$

Make a change of index in the m sum by substituting $n = m - r$ (so $m = n + r$). With the appropriate change in the summation limits and a bit of algebra, we obtain

$$\begin{aligned} W_k &= \sum_{r=0}^{N-1} \sum_{n=-r}^{N-1-r} e^{-2\pi ik(n+r)/N} x_r y_n \\ &= \left(\sum_{r=0}^{N-1} e^{-2\pi ikr/N} x_r \right) \left(\sum_{n=-r}^{N-1-r} e^{-2\pi ikn/N} y_n \right), \end{aligned}$$

$$\begin{aligned} W_k &= \left(\sum_{r=0}^{N-1} e^{-2\pi ikr/N} x_r \right) \left(\sum_{n=0}^{N-1} e^{-2\pi ikn/N} y_n \right) \\ &= X_k Y_k. \end{aligned}$$

Observações

H scales, and maybe phase-shifts, the input sinusoid S_i

In essence, we have now two alternative representations:

- determine the effect of L on s_i by convolution with h: $s_i * h$
- determine the effect of L on s_i by multiplication with H: $S_i \cdot H$

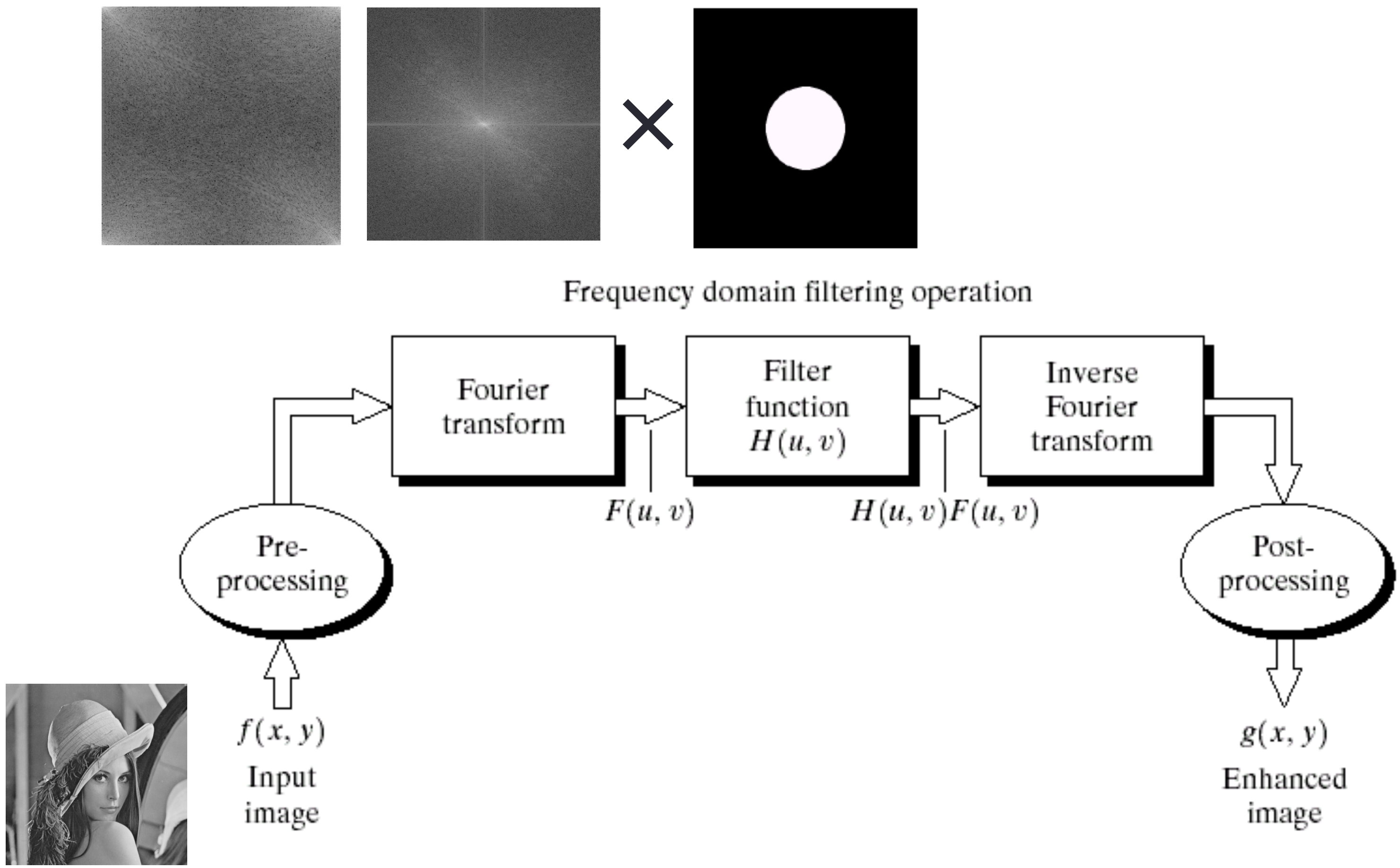
$$s_i * h \leftrightarrow S_i \cdot H$$

Since convolution is expensive for wide h, the multiplication may be cheaper

- but we need to perform the Fourier transforms of s_i and h

O processo de filtragem então baseia-se na confecção do vetor de amostras h, com DFT H

Filtragem no domínio de frequência



Design de filtro passa-baixa

- Design issue
  - G(u,v)=F(u,v) H(u,v)
  - Remove high freq. component (details, noise, …)
- Ideal low-pass filter
- Butterworth filter
- Gaussian filter

More smooth in the edge of cut-off frequency

Filtro passa-baixa ideal

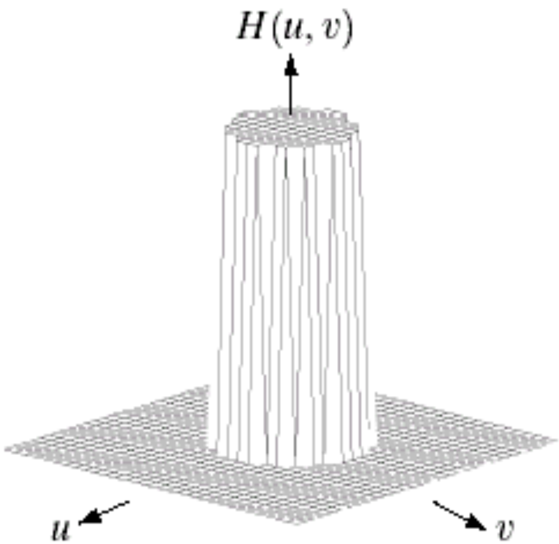
- Sharp cut-off frequency

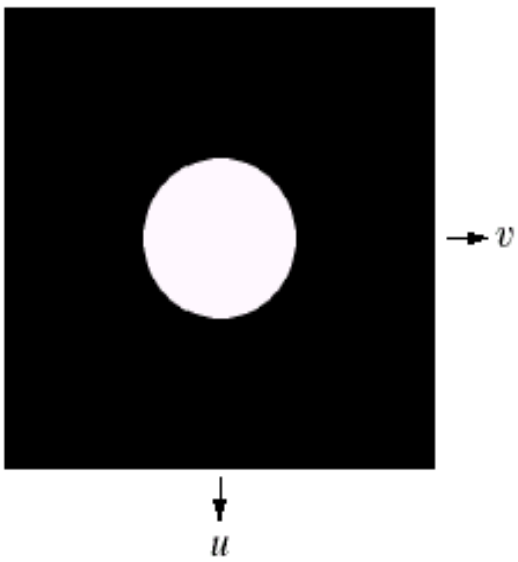
$$H(u,v)=\begin{cases}1 & \text{if } D(u,v)\le D_0\\ 0 & \text{if } D(u,v)\ge D_0\end{cases}$$

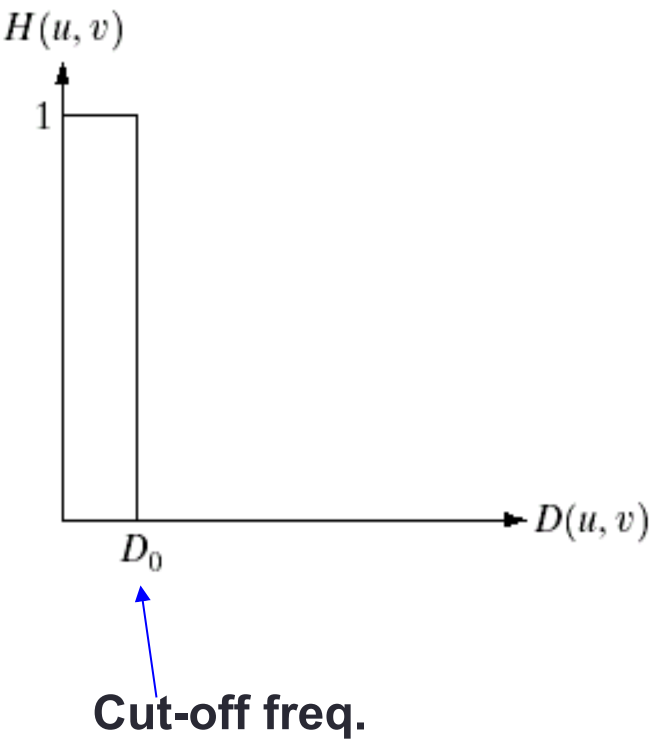
where D(u,v) is the distance to the center freq.

$$D(u,v)=[(u-M/2)^2+(v-N/2)^2]^{1/2}$$

Filtro passa-baixa ideal







Filtro passa-baixa ideal

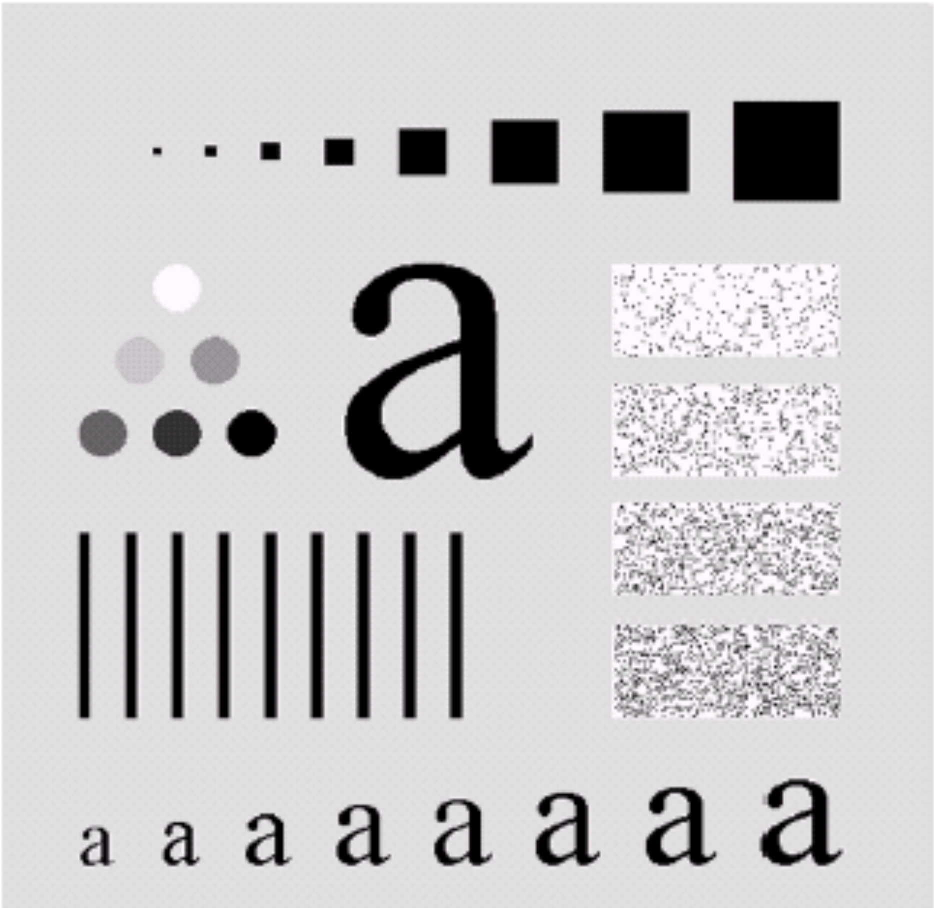
- ILPF can not be realized in electronic components, but can be implemented in a computer
- Decision of cut-off frequency?
  - Measure the percentage of image power within the low frequency

$$\alpha = 100 \times [\sum_{(u,v) \in cut-off \;\; freq} P(u,v)] / P_T$$

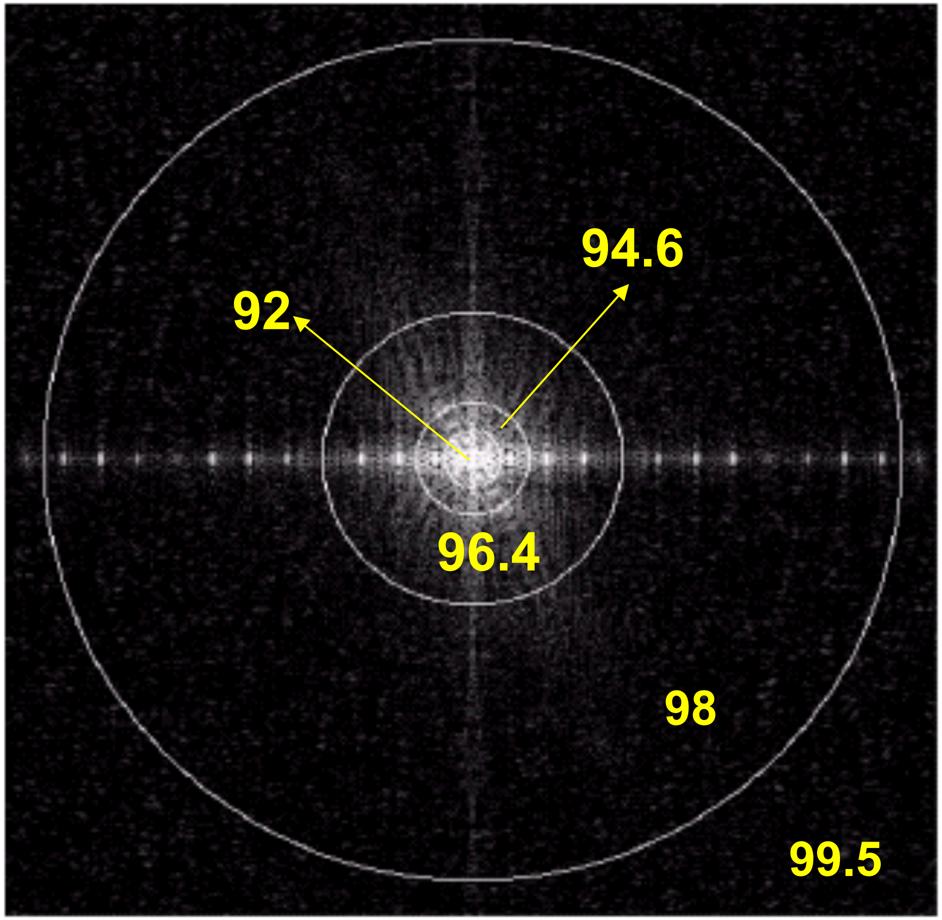
Total image power $$P_T = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} P(u,v)$$

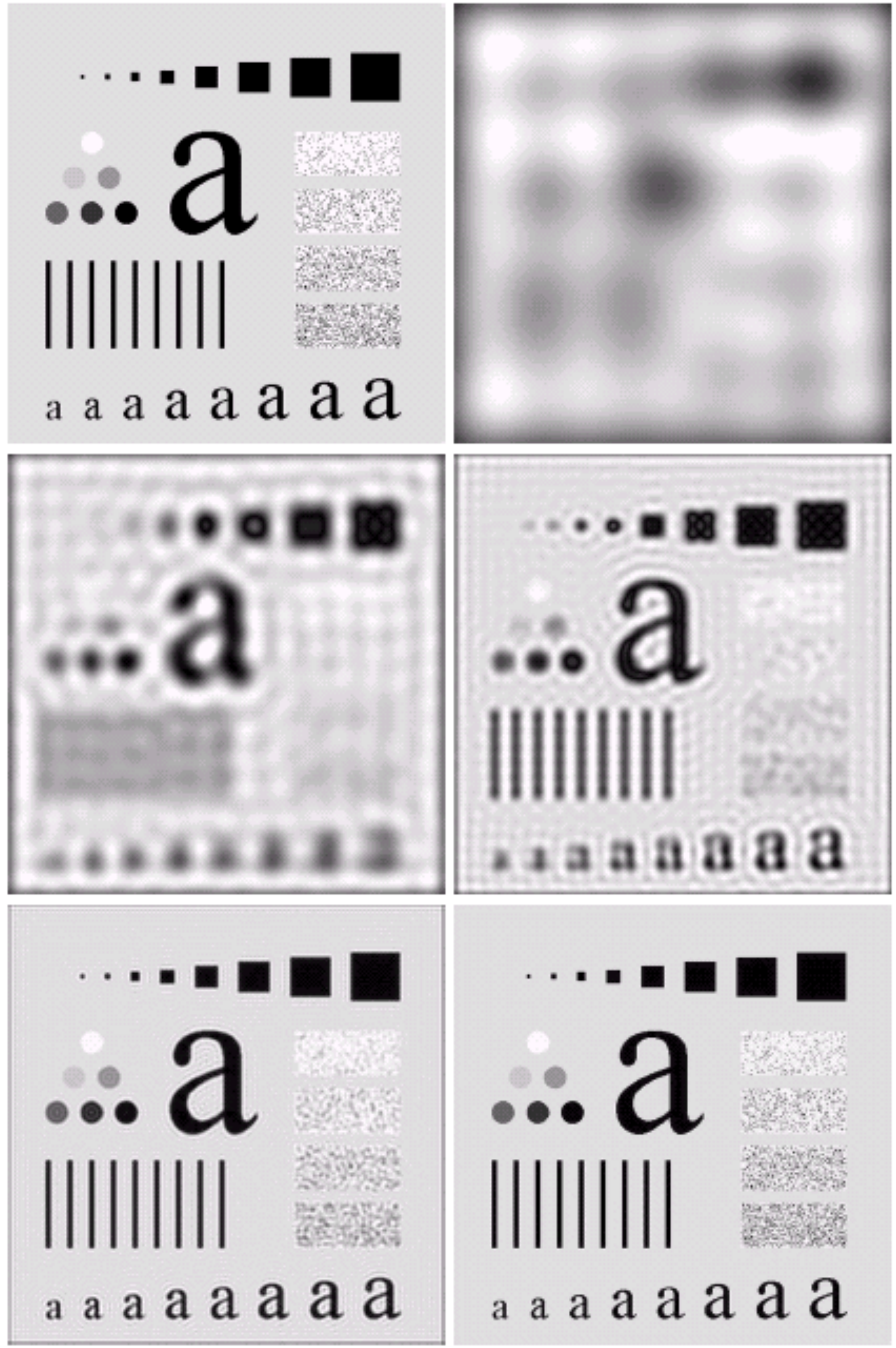
Distribuição de potência

original



frequências





original

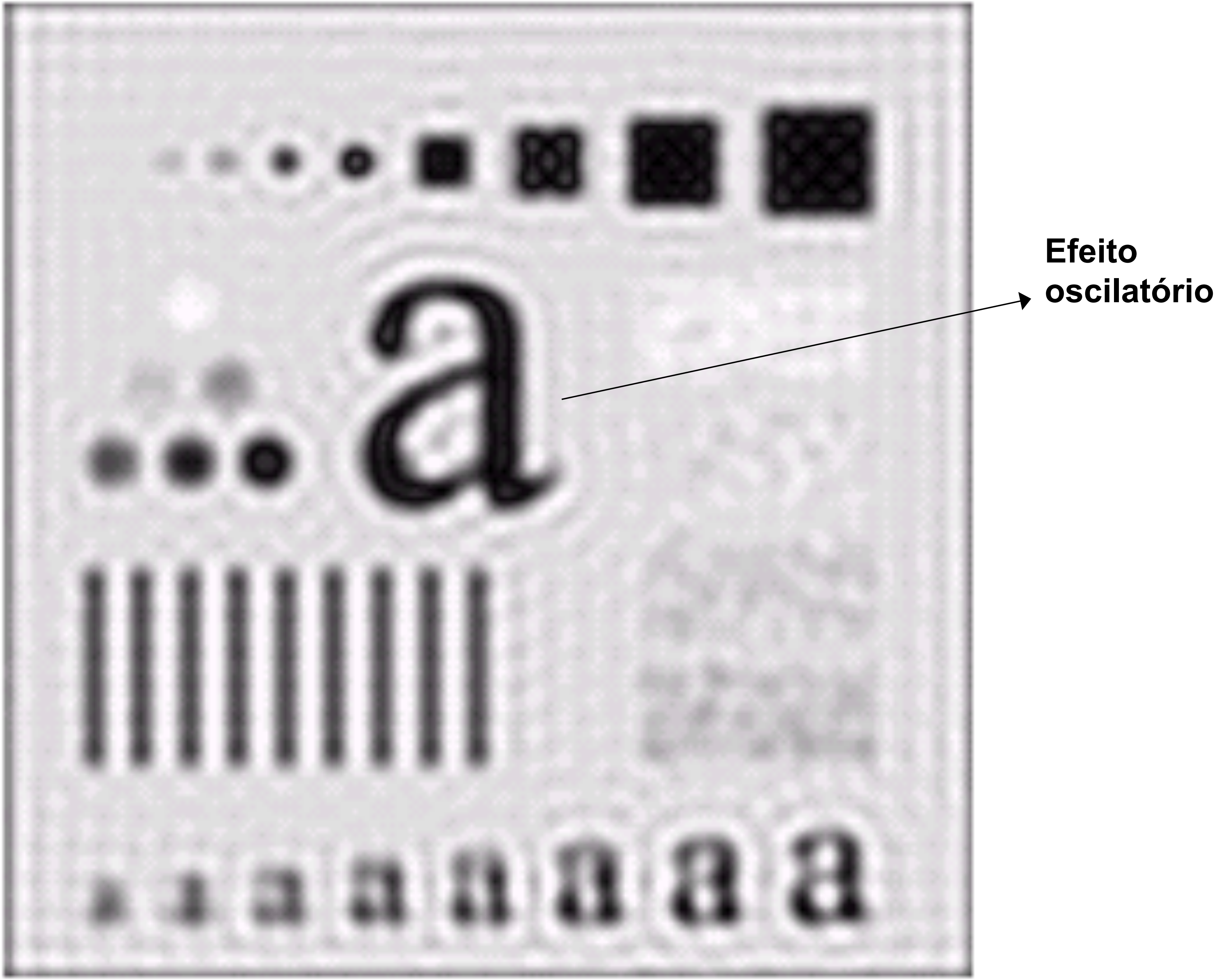
α=92
D_0=5

α=94.6
D_0=15

α=96.4
D_0=30

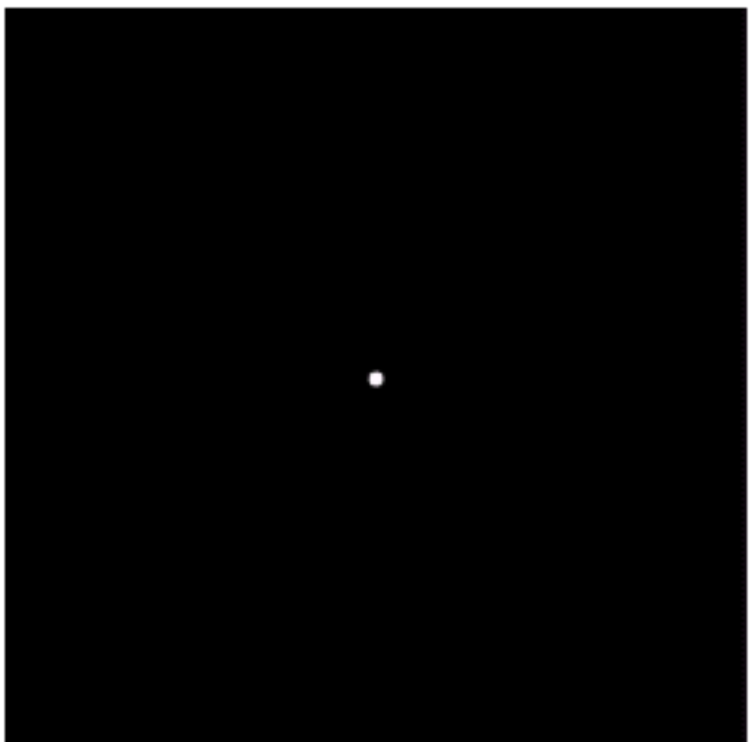
α=98
D_0=80

α=99.5
D_0=230

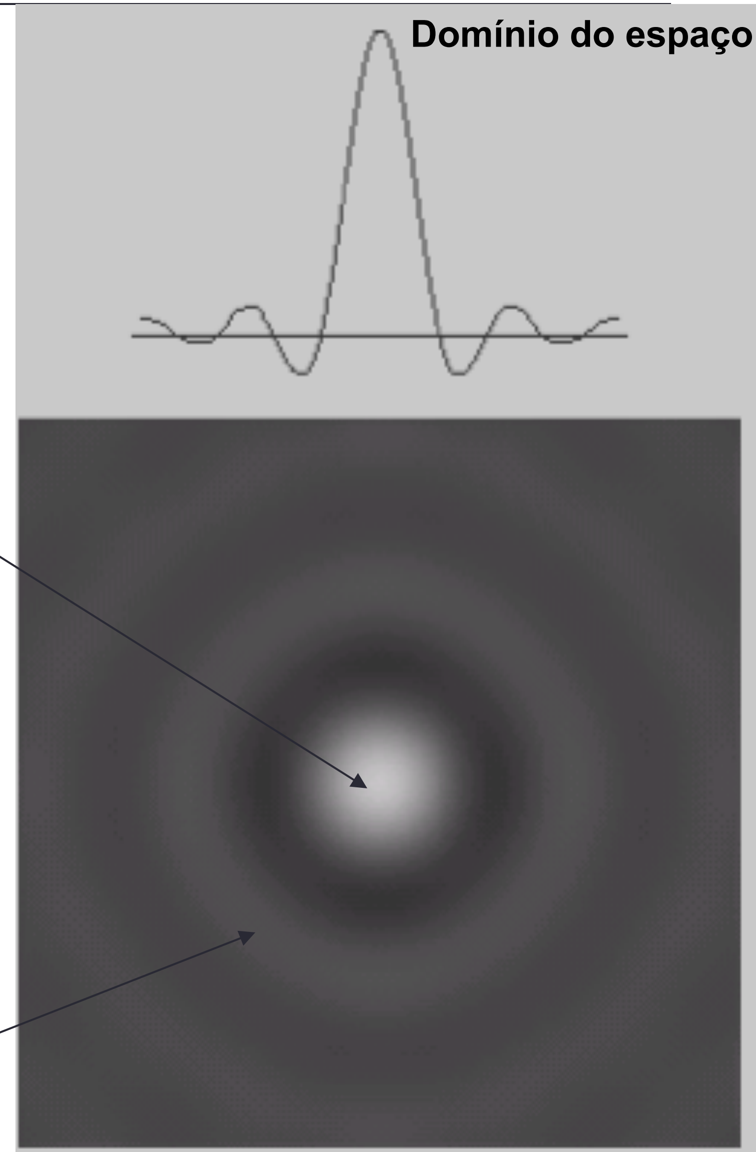

Efeito oscilatório

Efeitos do filtro passa-baixa

Domínio de frequências



Domínio do espaço



borramento

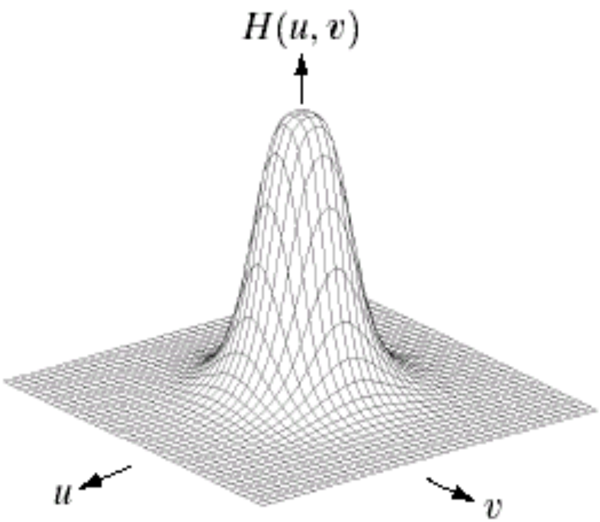
$\xrightarrow{F^{-1}}$

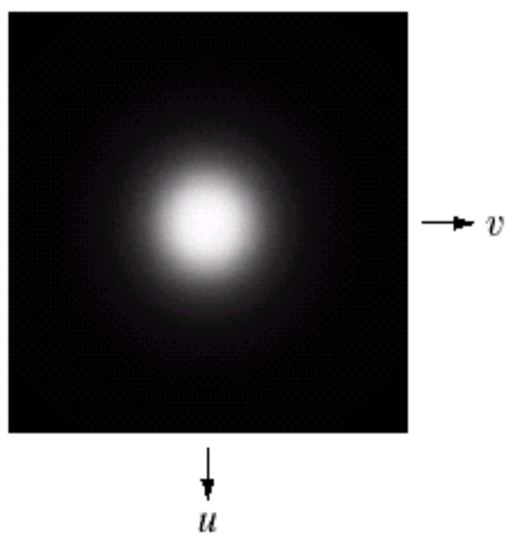
oscilação

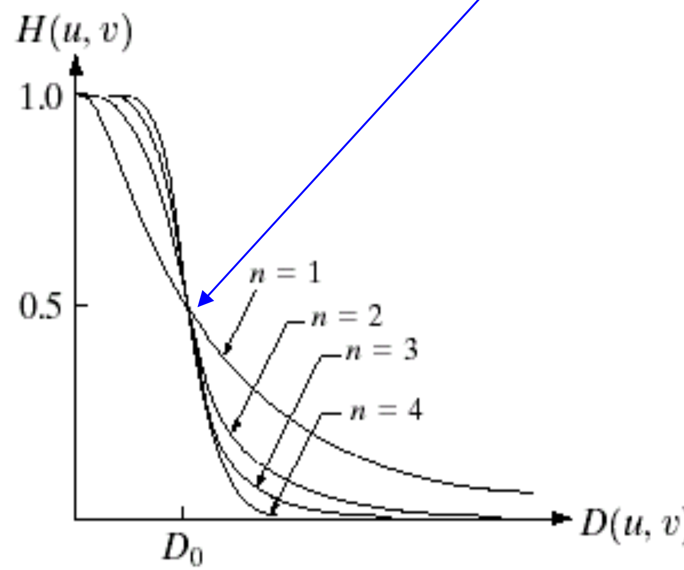
Filtros passa-baixa Butterworth

$$H(u,v) = \frac{1}{1 + [D(u,v)/D_0]^{2n}}$$

H=0.5 when $D(u,v)=D_0$

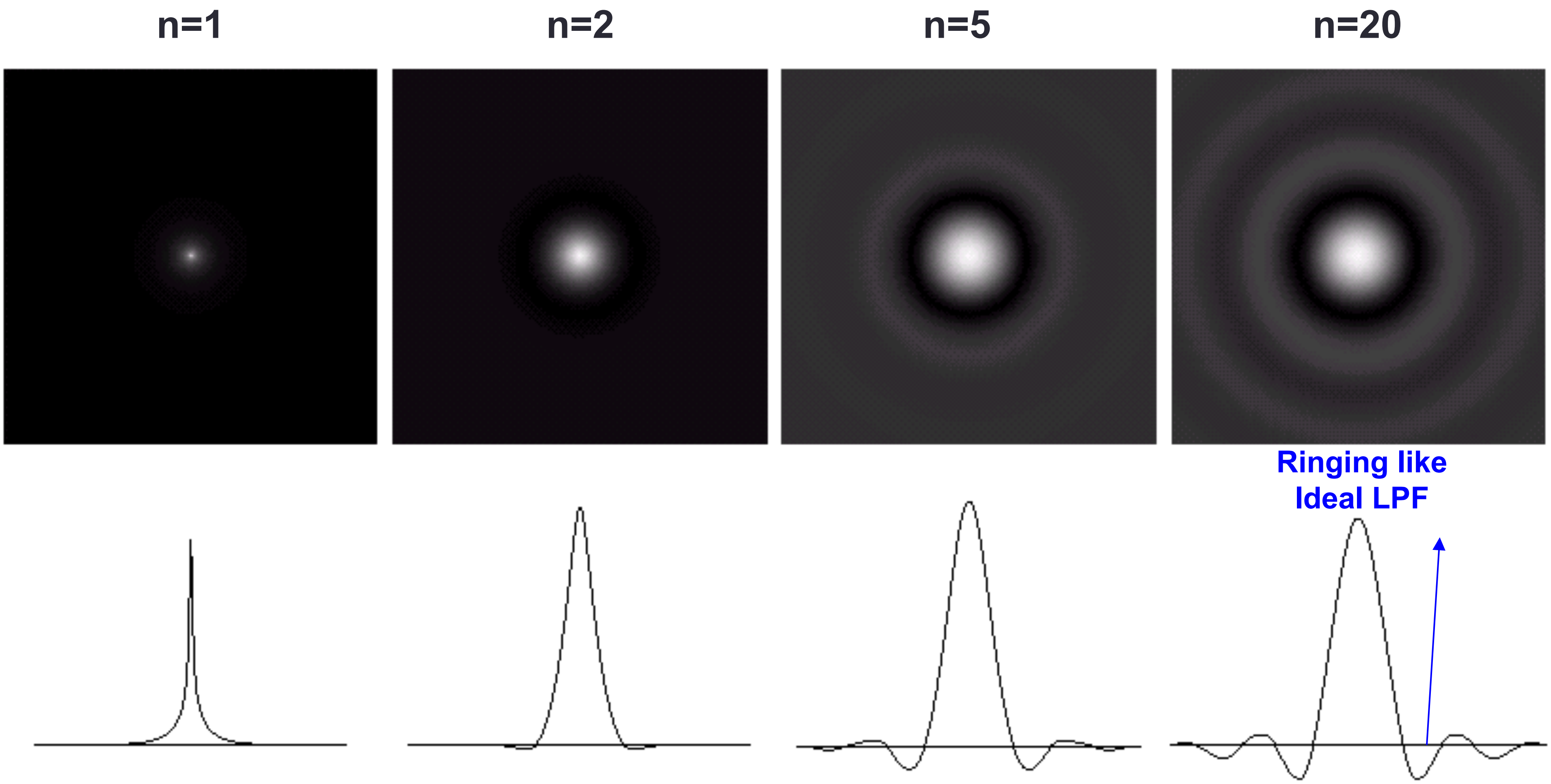






Ordem do filtro Butterworth

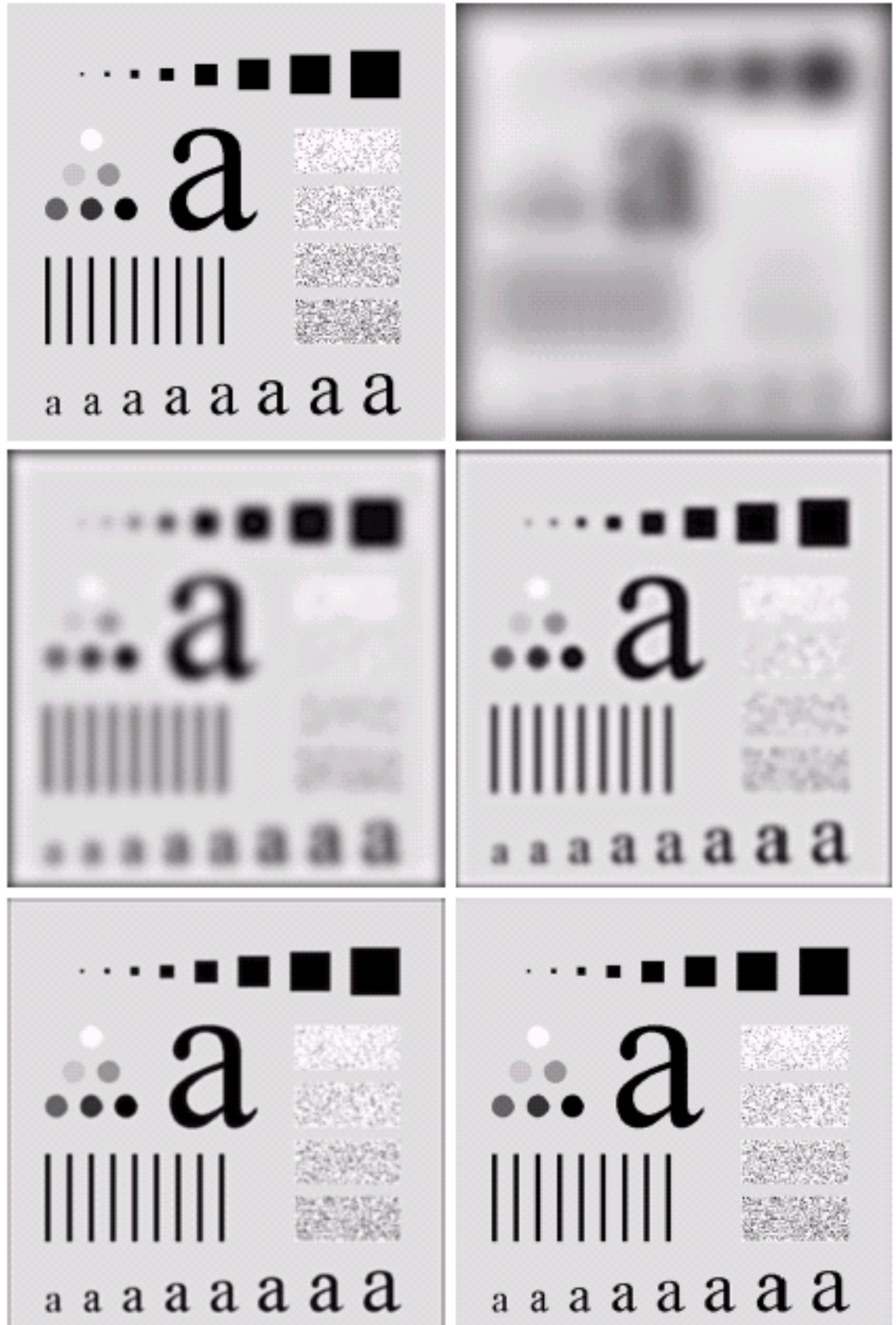
Spatial domain filter of butterworth filters



Ordem 2

original

$D_0=5$



$D_0=15$

$D_0=30$

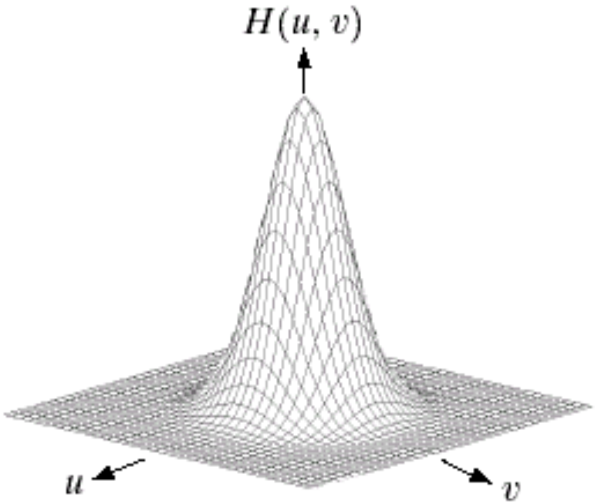
$D_0=80$

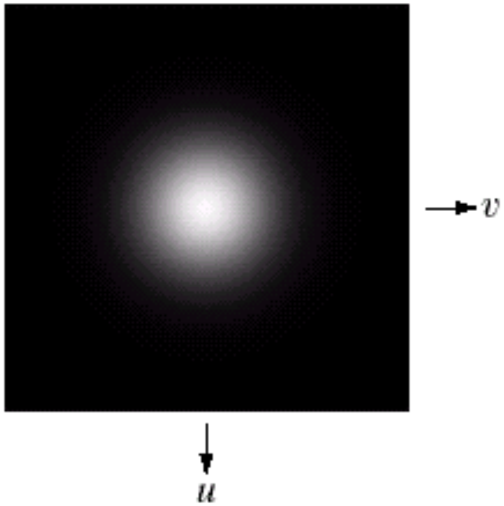
$D_0=230$

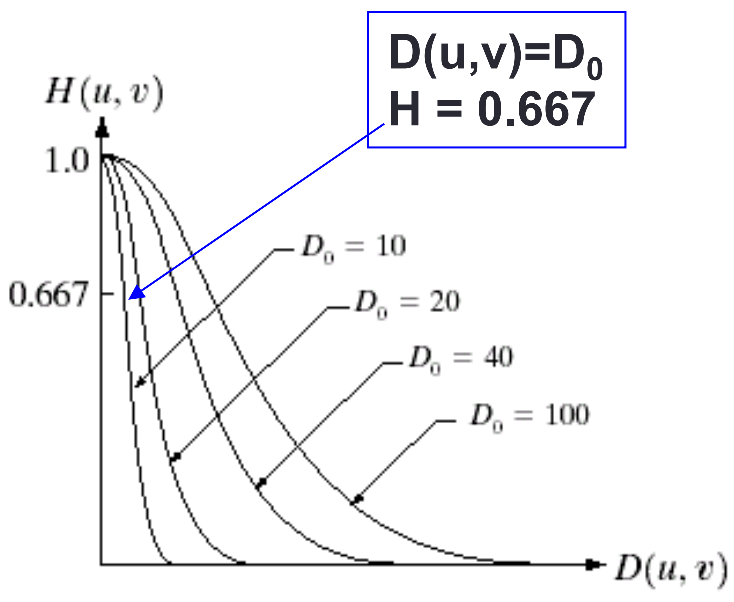
Filtro passa-baixa Gaussiano

$$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

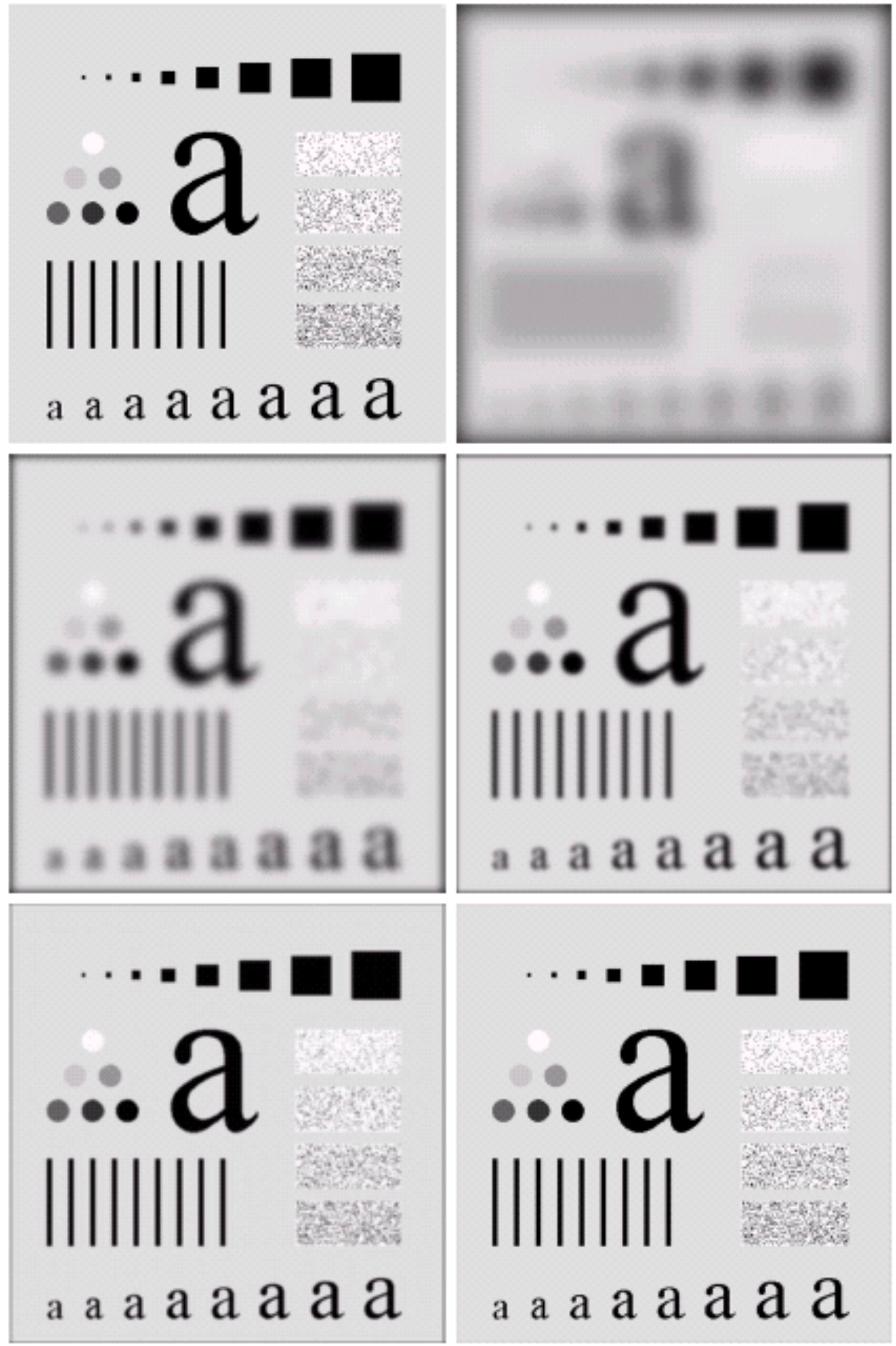
Variance or
cut-off freq.







Filtro Gaussiano



original

D_0=5

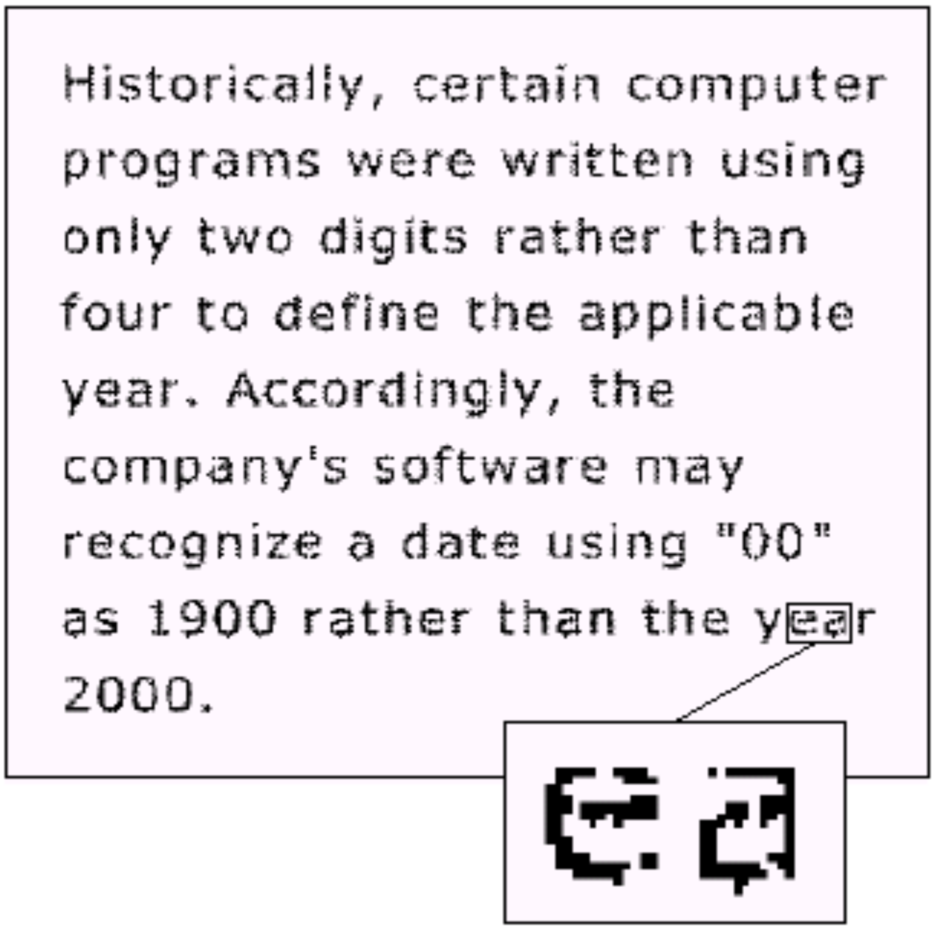
D_0=15

D_0=30

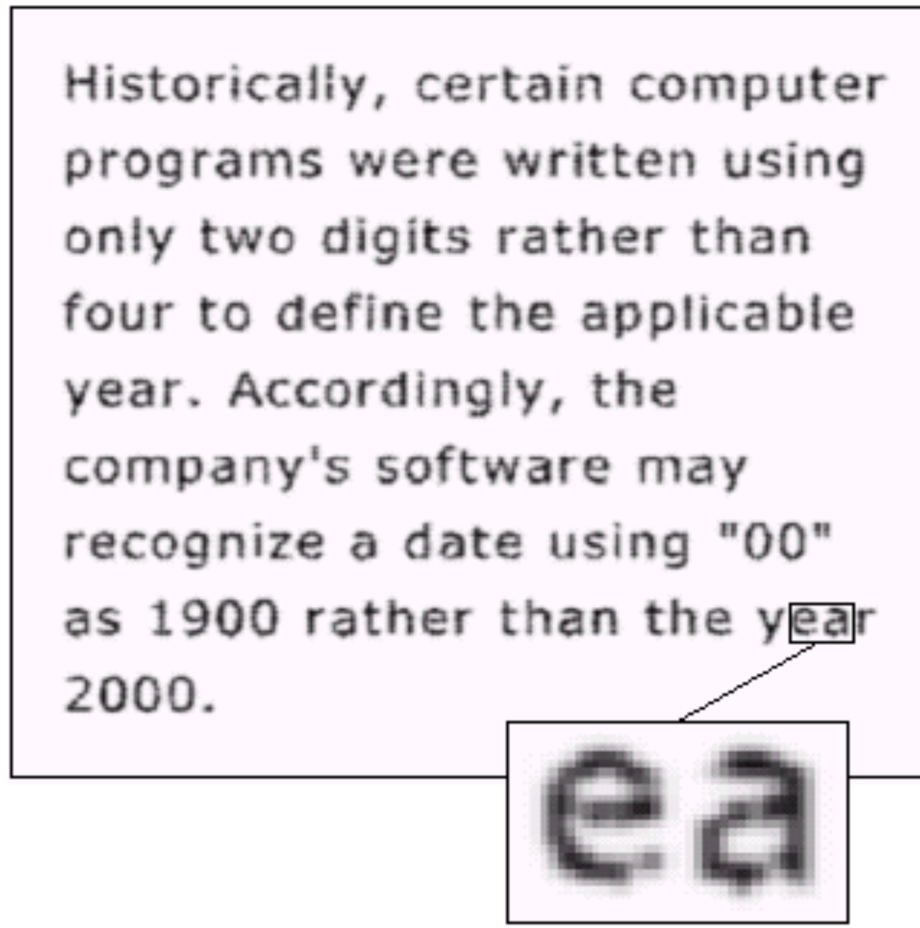
D_0=80

D_0=230

Aplicações práticas (i)



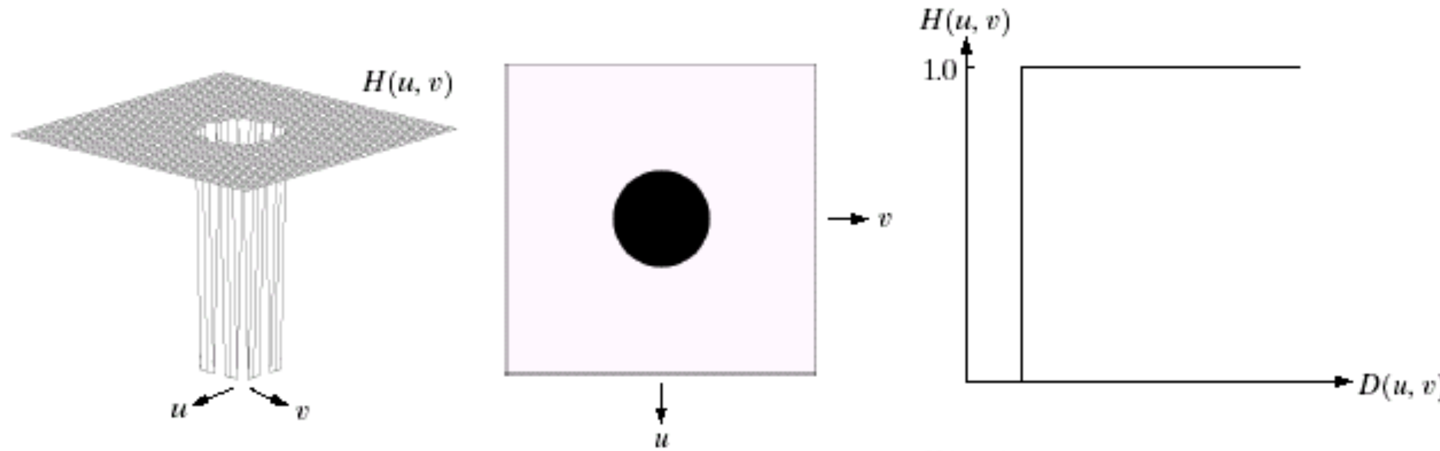
GLPF, D_0=80



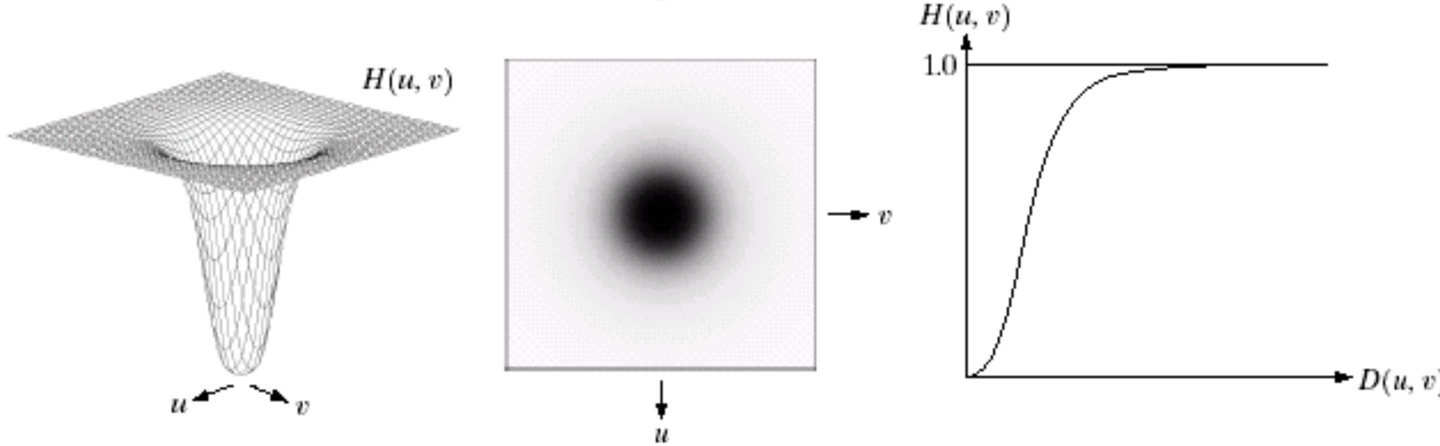
Filtros passa-alta

- Image details corresponds to high-frequency
  - Sharpening: high-pass filters
  - $H_{hp}(u,v)=1-H_{lp}(u,v)$
- Ideal high-pass filters
- Butterworth high-pass filters
- Gaussian high-pass filters
- Difference filters
  - Laplacian filters

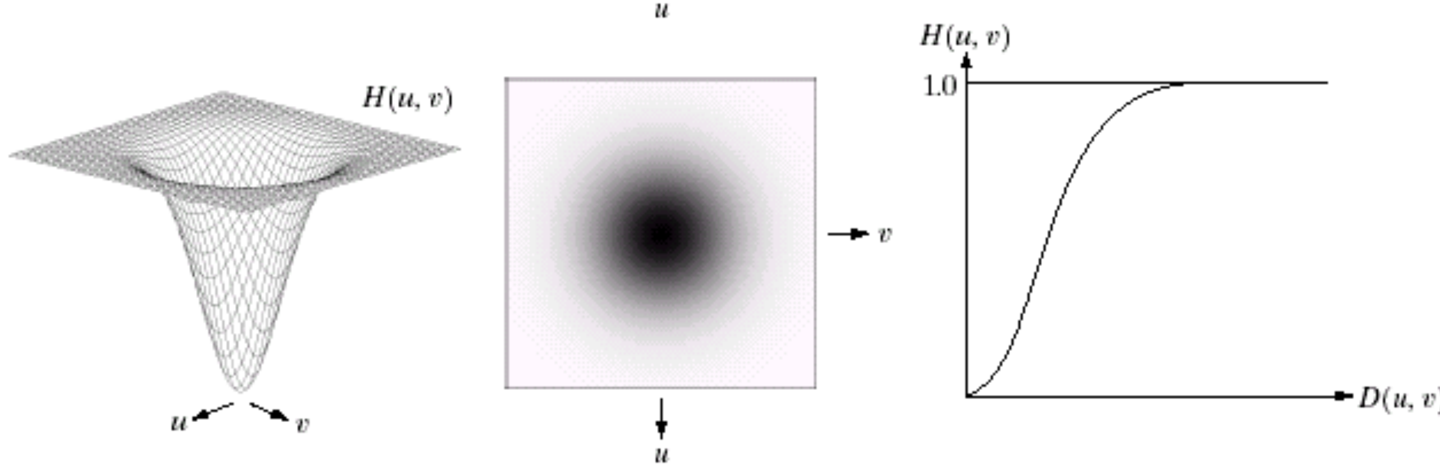
Ideal HPF



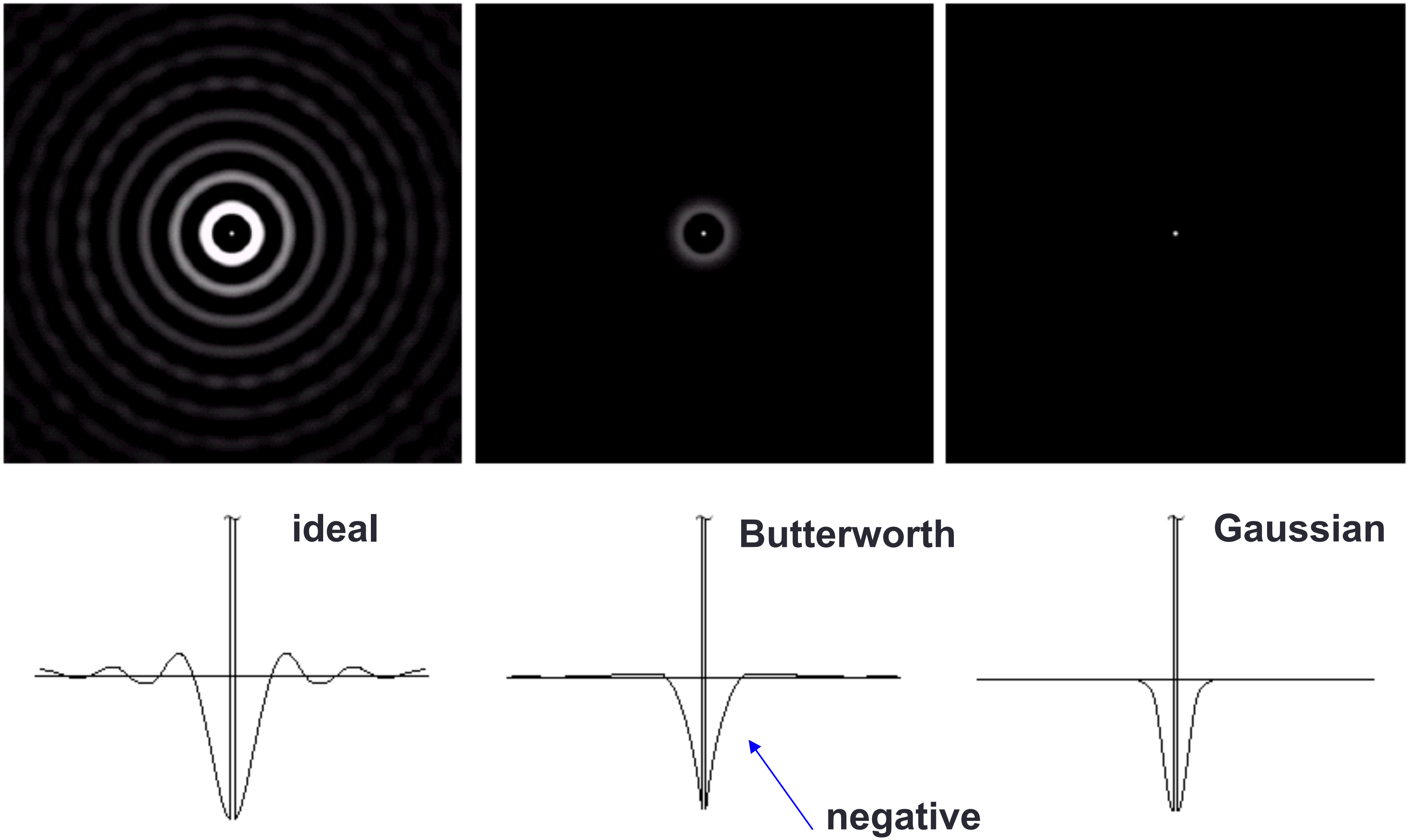
Butterworth HPF



Gaussian HPF



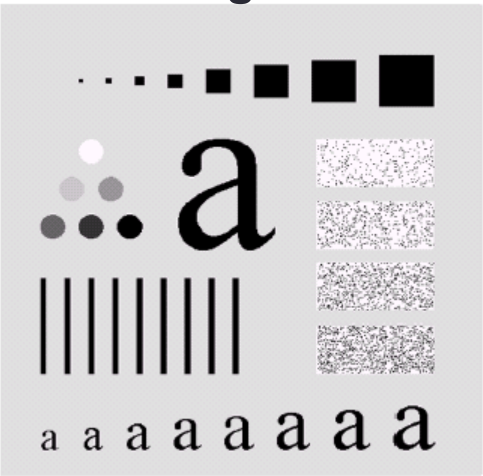
Filtros passa-alta (domínio do espaço)



Filtros passa-alta

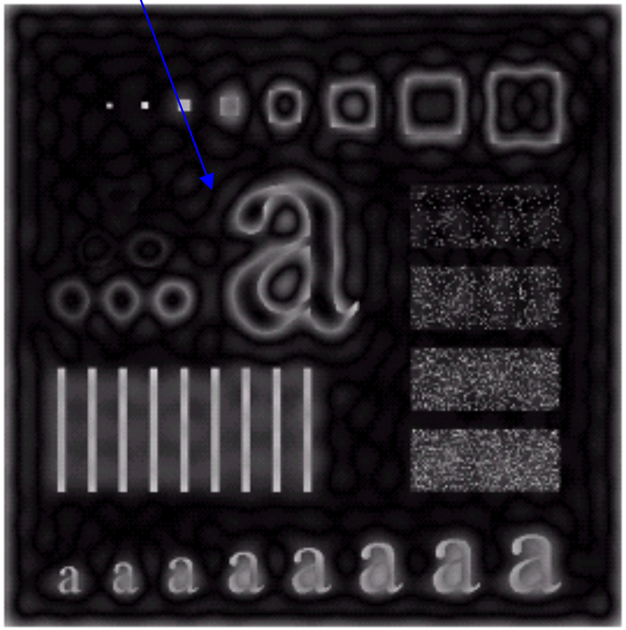
$$H(u,v)=\begin{cases}0 & \text{if } D(u,v)\le D_0\\ 1 & \text{if } D(u,v)\ge D_0\end{cases}$$

original

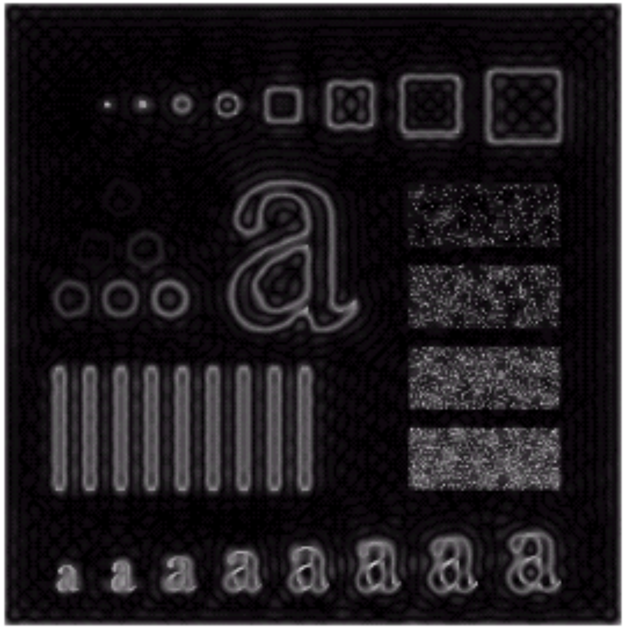


ringing

D_0=15



D_0=30

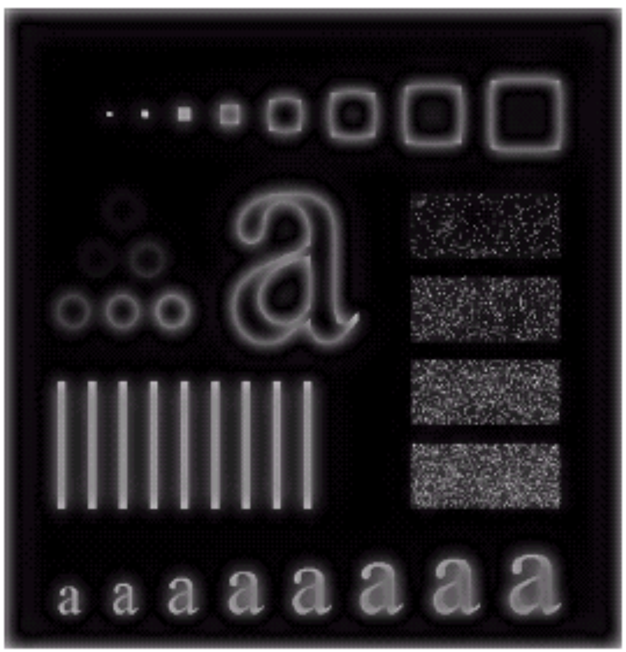


D_0=80

Filtros passa-alta Butterworth

$$H(u,v) = \frac{1}{1 + [D_0 / D(u,v)]^{2n}}$$

$n=2$, $D_0=15$



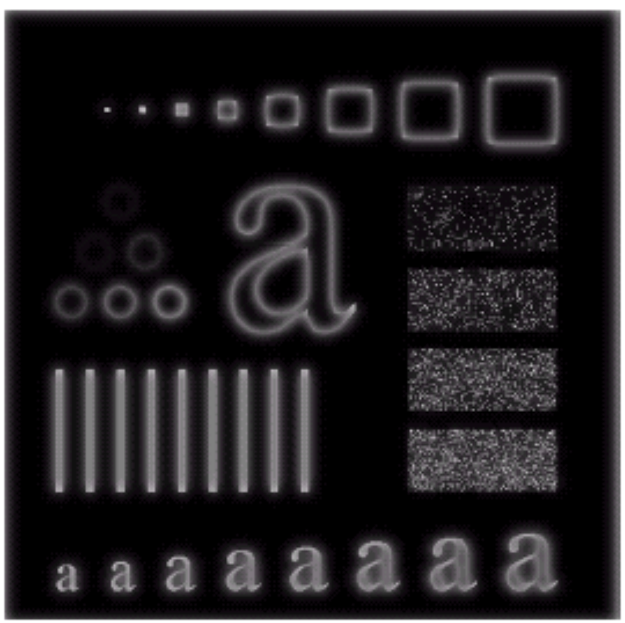
$D_0=30$

$D_0=80$

Gaussian high-pass filters

$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

D_0=15



D_0=30

D_0=80

Filtro Laplaciano

- Spatial-domain Laplacian (2nd derivative)

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

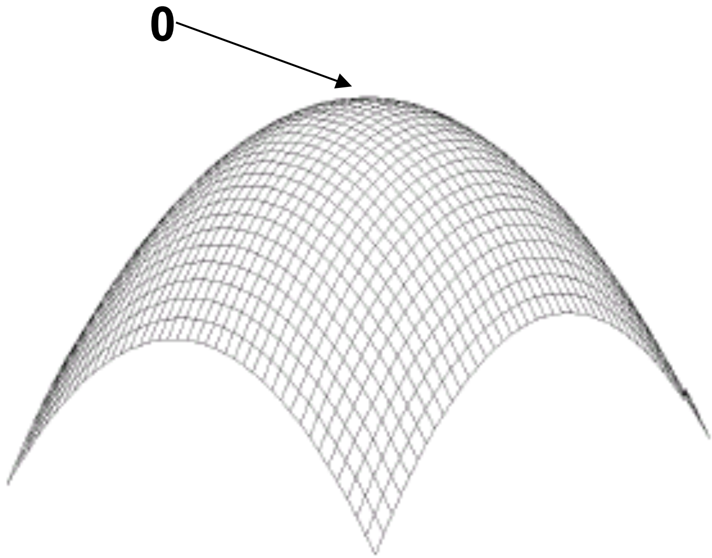
- Fourier transform

$$\Im\left[\frac{\partial^n f(x)}{\partial x^n}\right] = (ju)^n F(u)$$

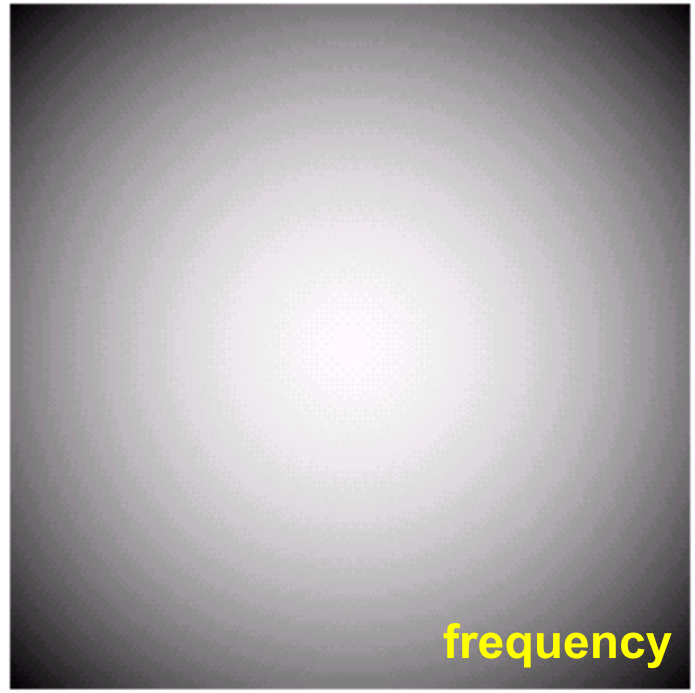
$$\Im\left[\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}\right] = (ju)^2 F(u,v) + (jv)^2 F(u,v)$$

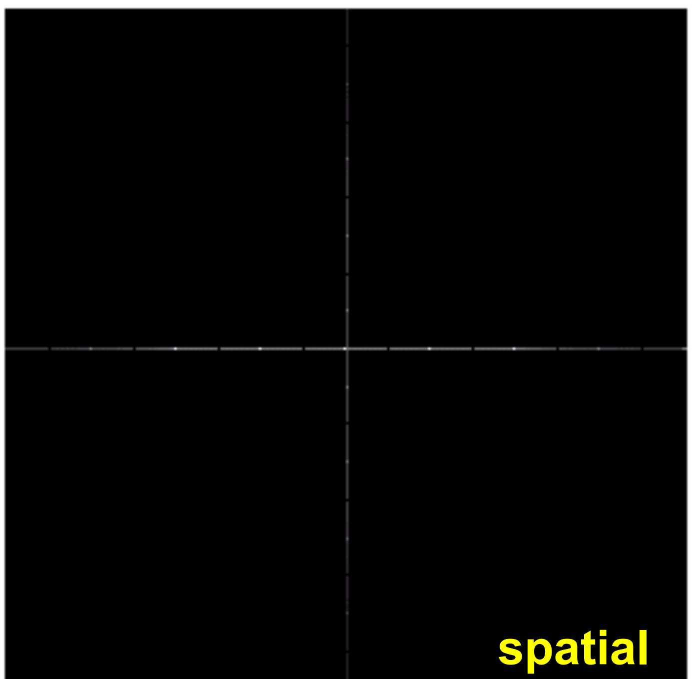
$$= \boxed{-(u^2 + v^2)} F(u,v)$$

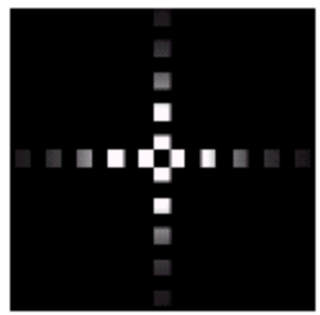
H(u,v) = -(u²+v²)

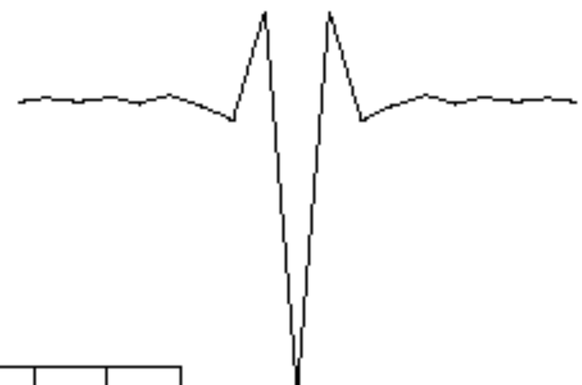


$$H(u,v) = -(u^2+v^2)$$

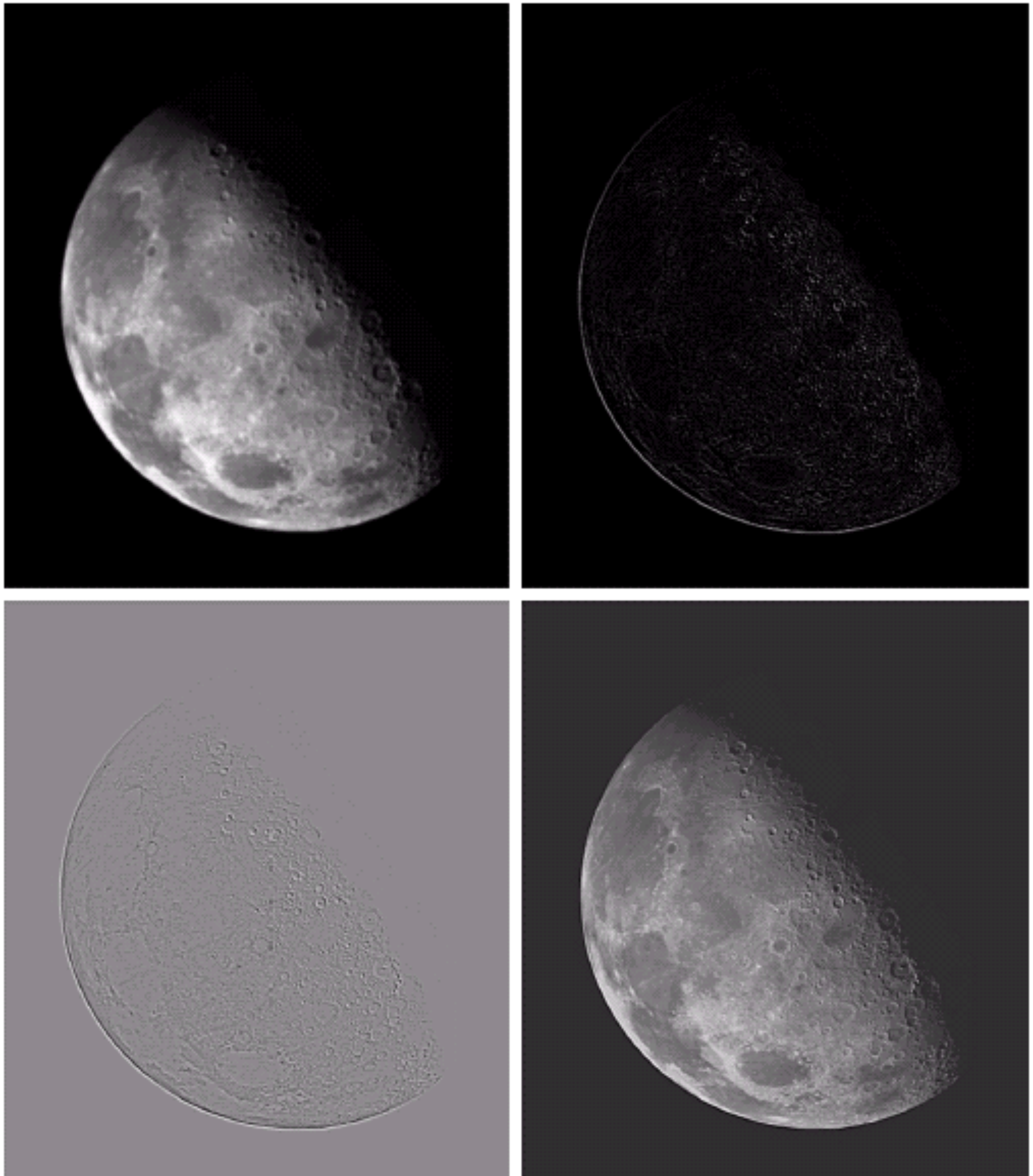








0	1	0
1	−4	1
0	1	0



original

Laplacian

Scaled
Laplacian

original+
Laplacian

Tarefa de casa

- Leitura do livro-texto capítulo 4 e exercícios