Spektrala Transformer för Media

Filtrering och transformer i 2D

Linjär bildbehandling

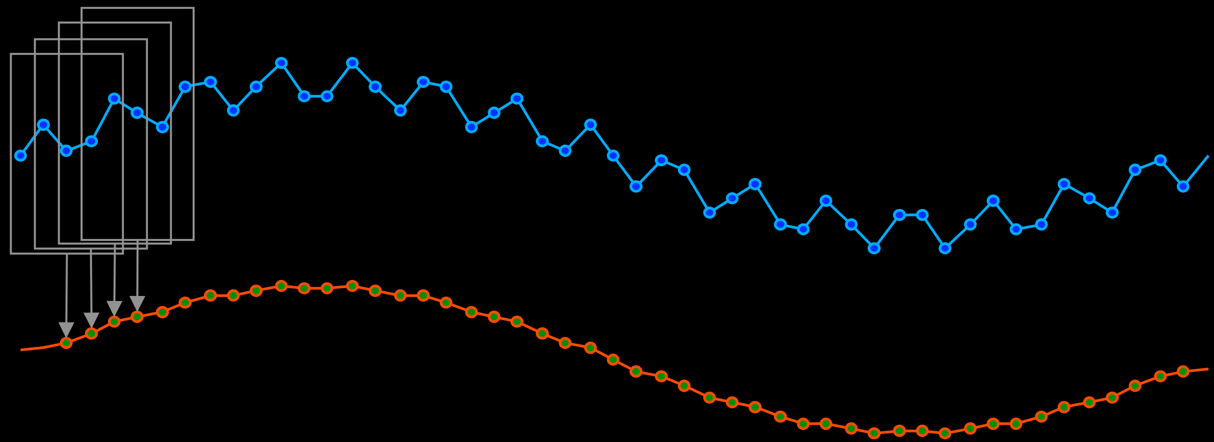
Principerna från 1-dimensionell signalbehandling kan appliceras även på 2D-signaler

Tillämpningar:

- Bildförbättring (brusreducering)
- Kant-detektion
- Omsampling (anti-aliasing)
- Bildkomprimering (JPEG)

mm...

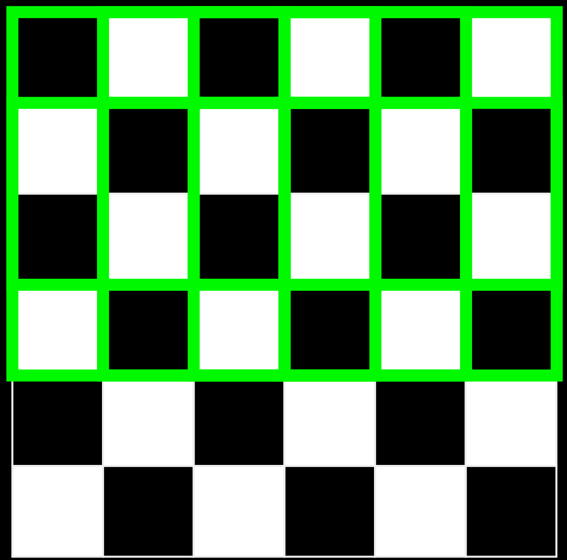
Filtrering i 1D

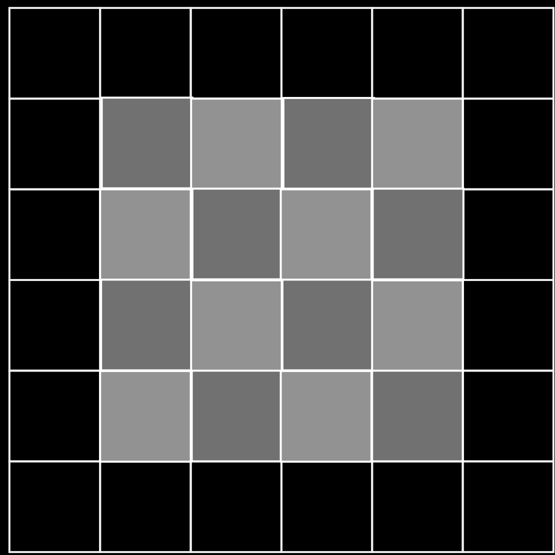


Exempel

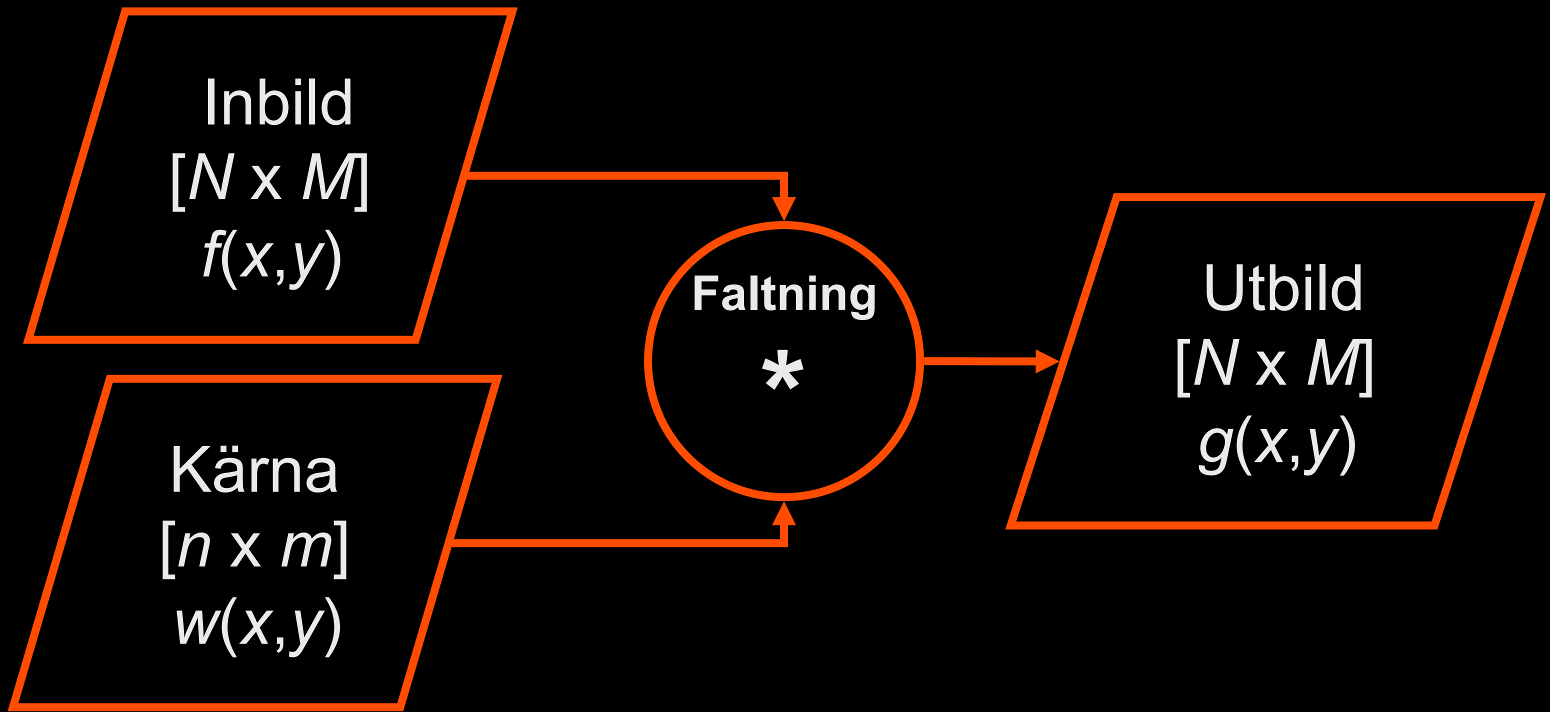
- Signal med högfrekvent störning
- Skapa ny signal genom att medelvärdesbilda över 5 punkter i taget
- Kallas moving-average / rullande medlevärde

Filtrering i 2D





Filtrering i 2D

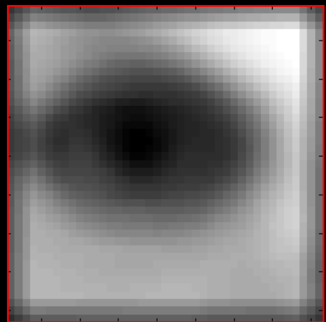


Filtrering i 2D – kärna, PSF

- Filtrering innebär faltning (convolution) med en kärna (kernel, mask)
- Kärnan är filtrets 2-dimensionella impulssvar
- Kallas även PSF (point spread function) – dvs. hur en punkt sprids av filtret

Filtrering i 2D: lågpass

- Lågpassfiltrering ger utjämning/ oskärpa (smoothing/ blur)
- Användning: brusreducering, nedsampling (anti-alias) mm.
- Motsvarar *integrering*
- Exempel: moving average (7x7)



1	1	1
1	1	1
1	1	1

1	2	1
2	4	2
1	2	1

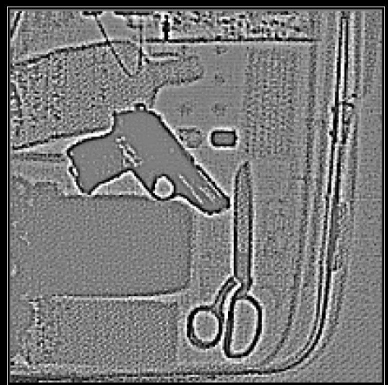
Matlab: `I2=conv2(I,ones(7));`

Filtrering i 2D: högpass

- Högpassfiltrering förstärker *kanter* i bilden (edge detection)
- Undertrycker jämna partier
- Motsvarar 2:a ordningens *derivering*

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1



Exempel: sharpening

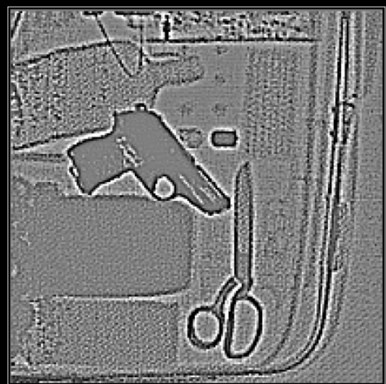
(sharpening, edge enhancement)

- Kombination av originalbild och högpass förstärker bildens kanter

0	0	0
0	1	0
0	0	0

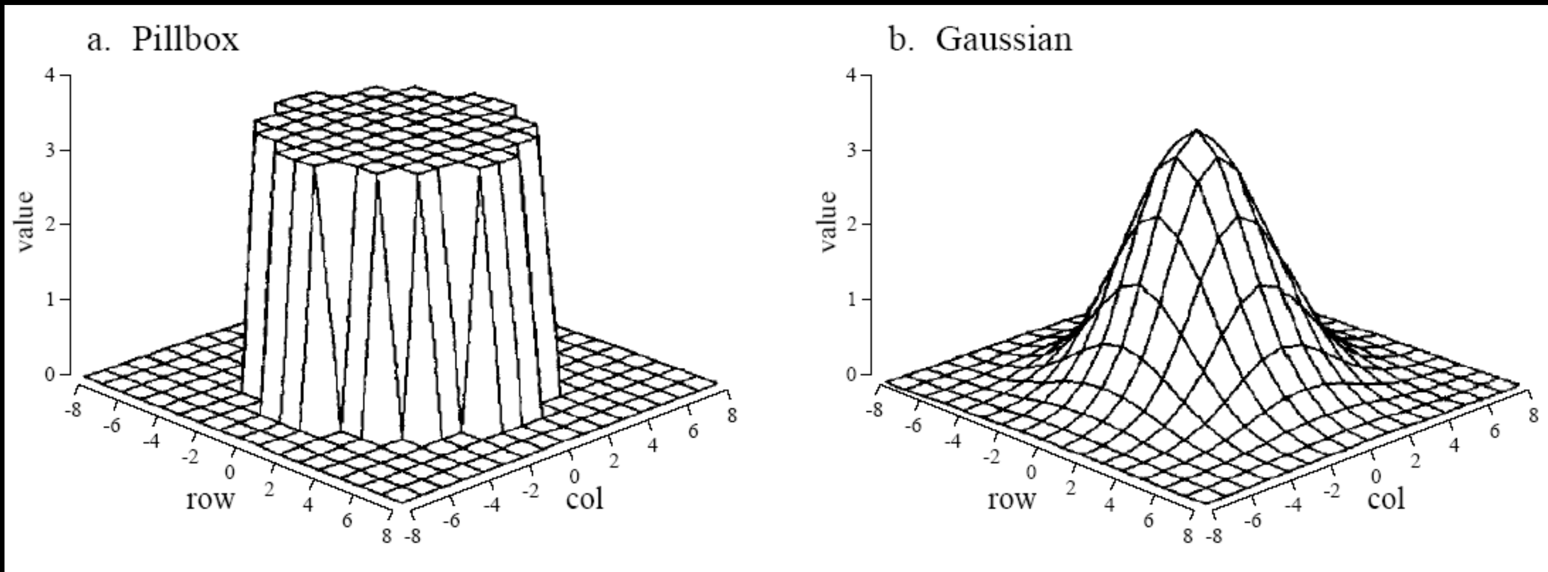
-1	-1	-1
-1	8	-1
-1	-1	-1

$-k$	$-k$	$-k$
$-k$	$1+8k$	$-k$
$-k$	$-k$	$-k$



Cirkulärt symmetrisk kärna

- En cirkulärt symmetrisk filterkärna är ofta önskvärd
- Modifierar bilden likformigt i alla riktningar



Faltning i 2D - beräkningar

N x *M* bild

n x *m* kärna

Kräver *N* x *M* x *n* x *m* multiplikationer

Exempel:

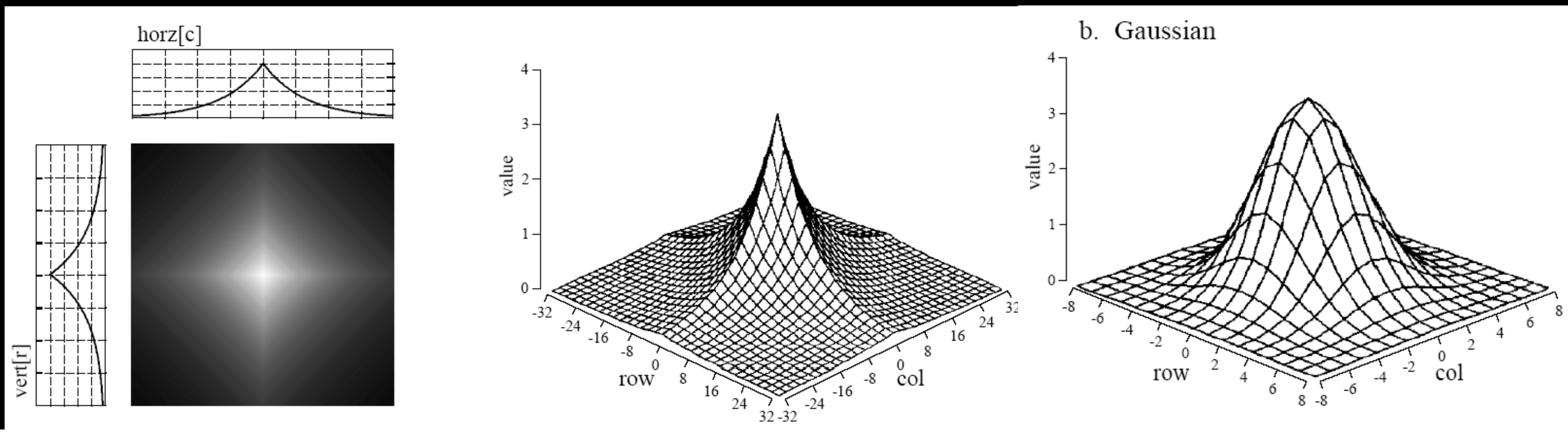
*1000 x 1000 bild * 100 x 100 kärna*

10^{10} mult.

Sökes: metod för snabbare faltning!

Separerbar filterkärna

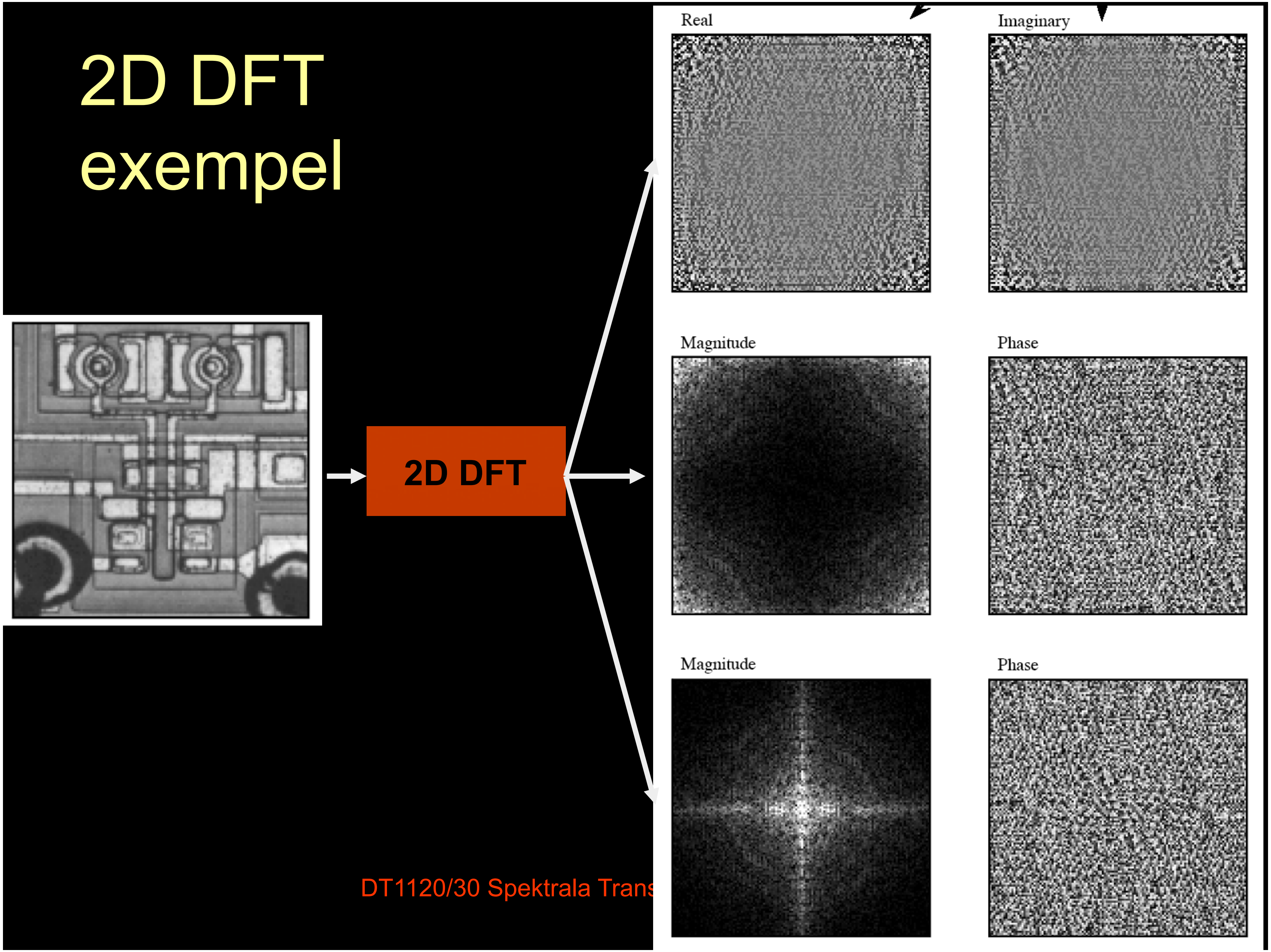
- Om filterkärnan $w(x,y) = g(x)h(y)$ så är den separerbar i x och y.
- Faltningen kan då göras med N x M x (n + m) multiplikationer
- Gauss-funktionen är *både* separerbar *och* cirkulärt symmetrisk



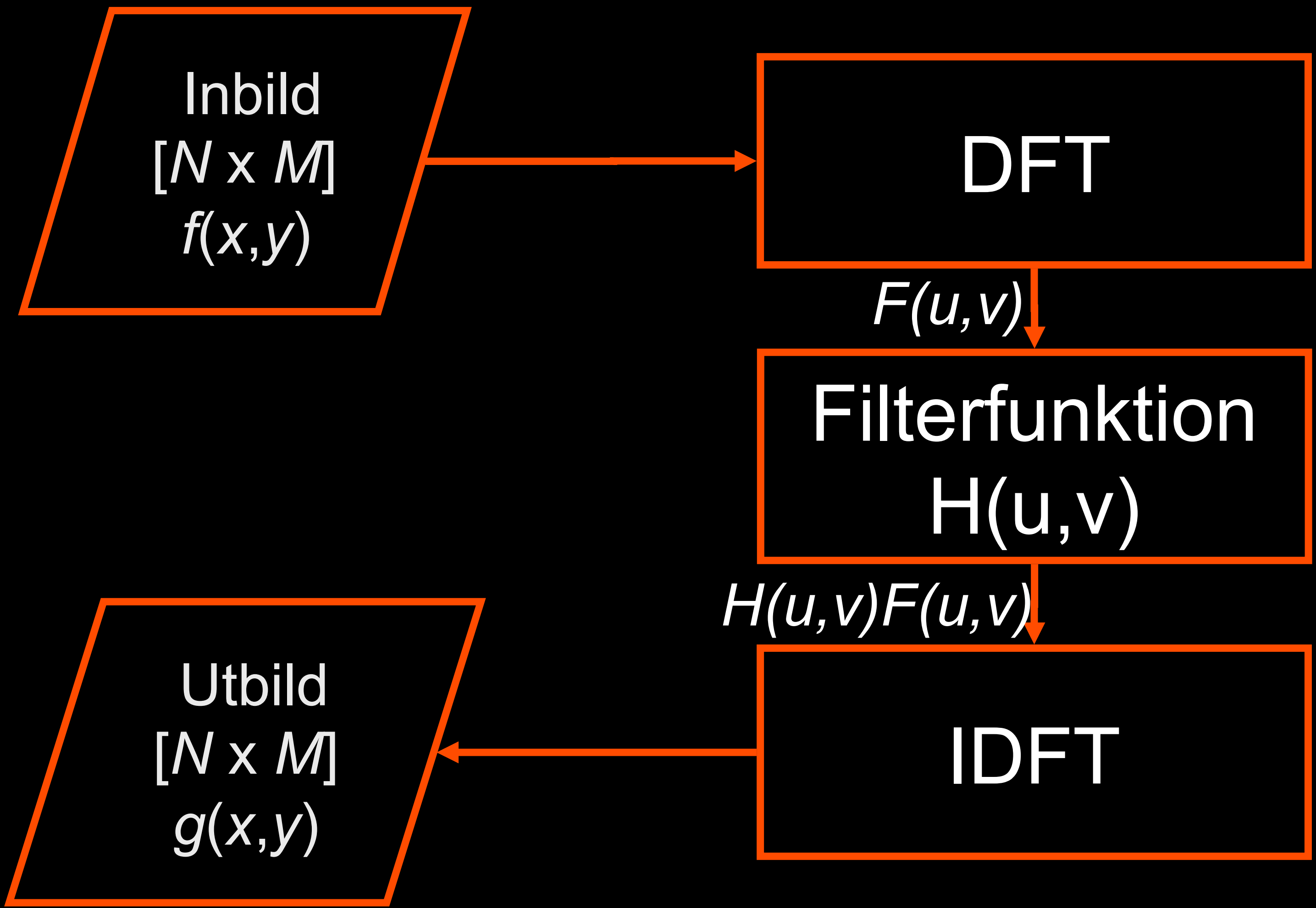
DFT i 2D

- 2D DFT beräknas med 1D FFT längs rader och kolumner (eller omvänt)

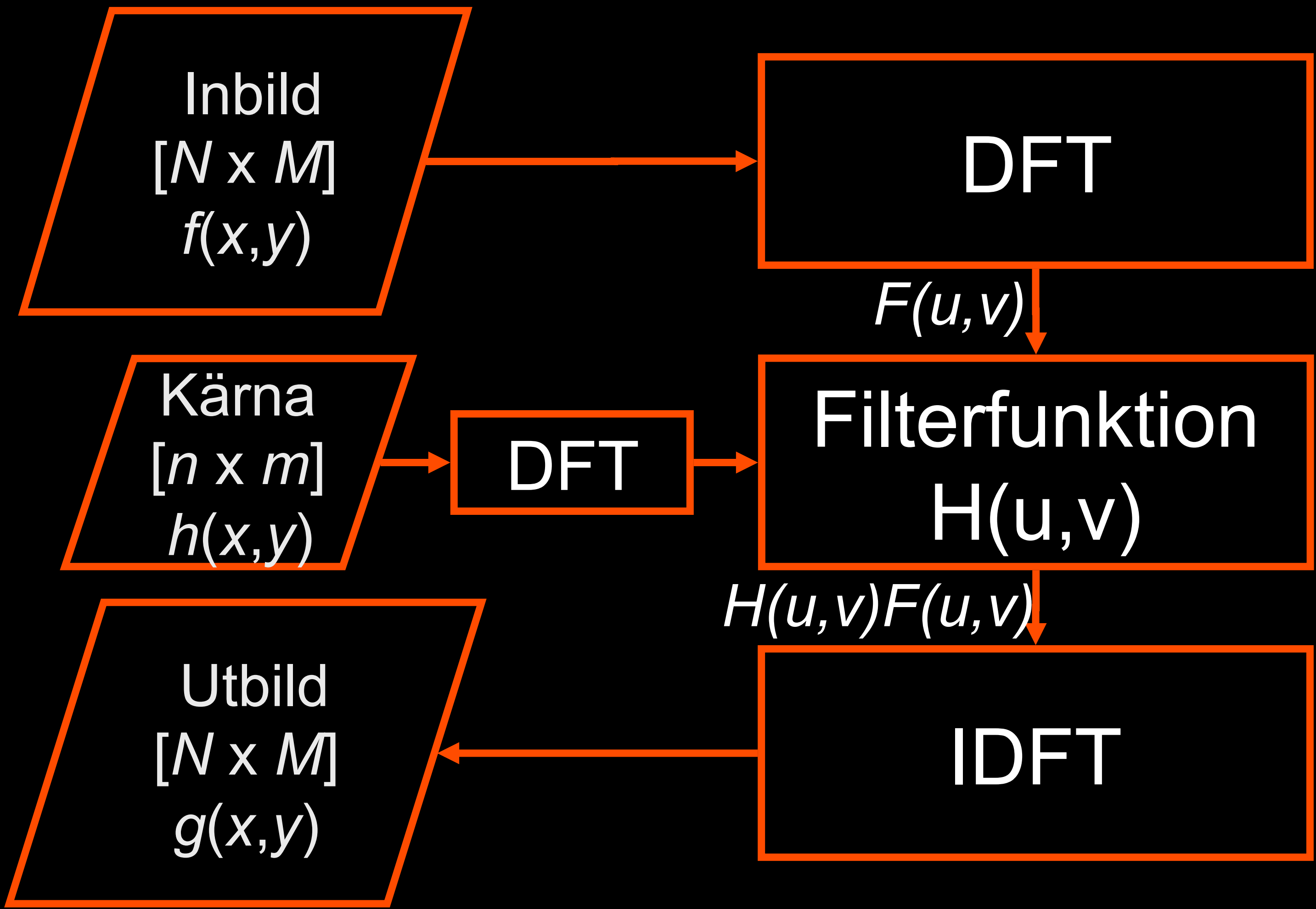
2D DFT exempel



Filtrering i frekvensdomänen



Filtrering i frekvensdomänen



Filtrering i frekvensdomänen

Bygger på faltningsteoremet (2D-upplagan):

Faltning i spatialdomänen

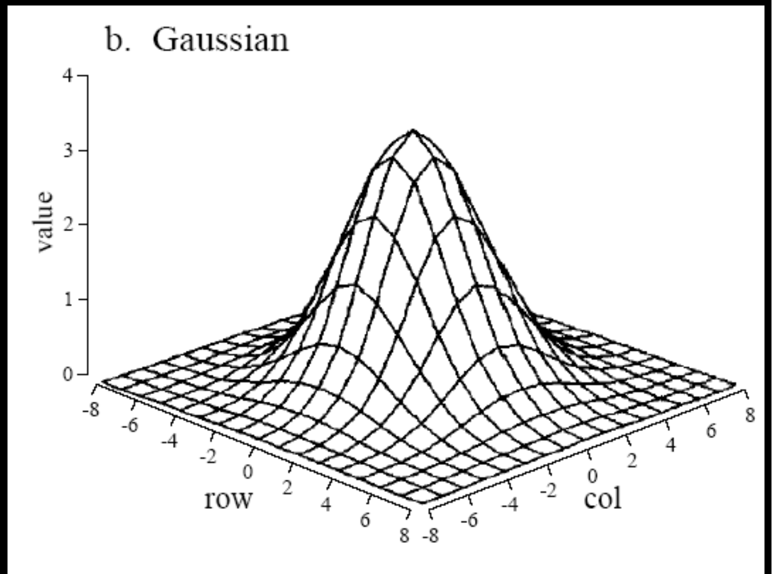
$\leftrightarrow$

Multiplikation i frekvensdomänen

$$f(x,y) * h(x,y) \Leftrightarrow F(u,v)H(u,v)$$

Gauss - lågpassfilter

- Gauss spektralt ↔ gauss spatialt
- Motsvarar spridningen i en lins
- Cirkulär symmetri
- Separerbar i x och y-led



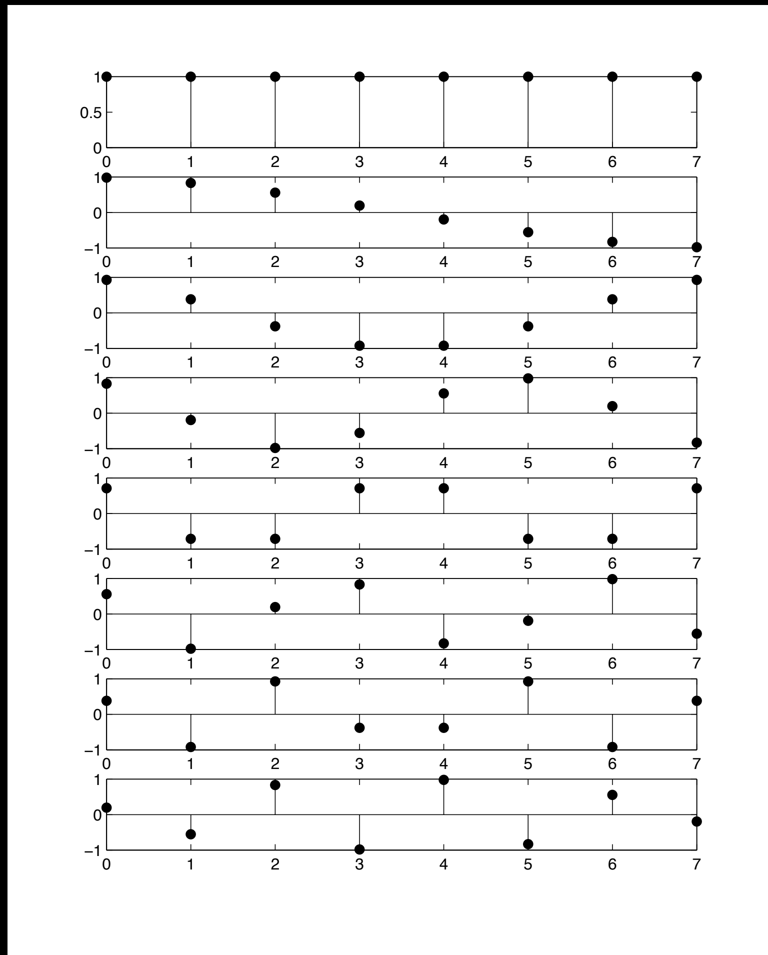
Discrete Cosine Transform (DCT)

- Nära släkting till DFT'n
- Endast cosinus basfunktioner
- Helt reell
- Beskriver icke-periodiska signaler bättre än DFT'n
- Används ofta för data-kompression
- MP3 (kodar ”koefficient-rörelser”)
- JPEG
- Kan beräknas effektivt ur FFT'n

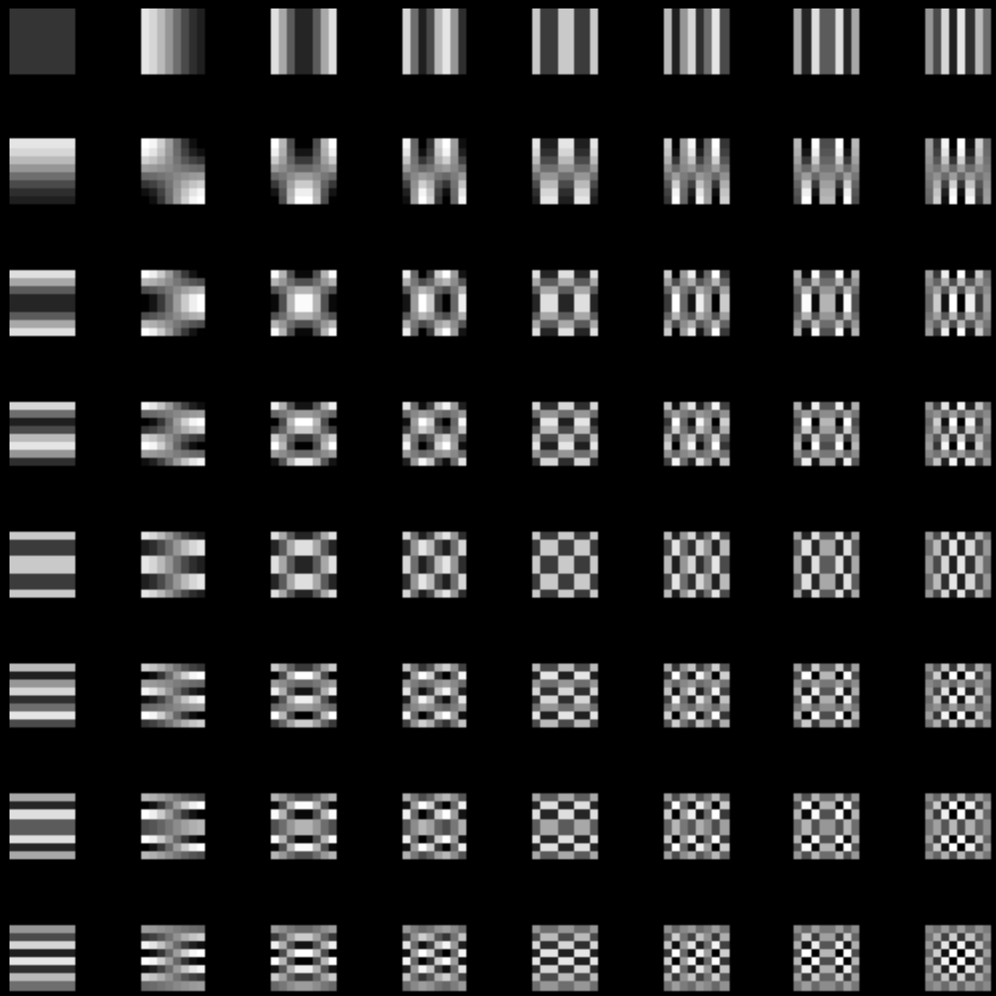
$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right]$$

DCT basfunktioner (N=8)

$$\cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right]$$



DCT basfunktioner i 2D (8x8)



Slutsatser

- 2D-filtrering är resurskrävande för icke-triviala filterkärnor
- Faltningen kan snabbas upp genom att använda separerbara filterkärnor
- Faltningen kan göras som en multiplikation i frekvensdomänen
- DCT-transformen är en DFT-släkting som används för att beskriva/komprimera icke-periodiska signaler