

Projekt 1.6 Wavelets

Indhold

Den digitale revolution	2
Øvelse 1: Eksempel på en fraktal graf	3
1. Digitale data.....	3
Øvelse 2 Fingeraftryk.....	4
2. Wavelet analyse.....	5
Øvelse 3: Konstruktion af wavelets	5
Øvelse 4: Haar-waveletten	5
Øvelse 5: Udregning af wavelet koefficienter	8
Øvelse 6 Skema over fadersøjler og modersøjler.....	9
3. Den inverse wavelet transformation	11
Øvelse 7: Wavelet-analyser	12
4. Fourier analyse	13
5. Digital Fourier transformation.....	14
Øvelse 8: Sinus wavelets.....	15
Øvelse 9: Diskret sinus-transformation	15
Øvelse 10: Fourieranalyser af simple signaler	16

Den digitale revolution

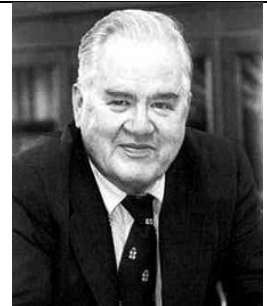
Computerens indtog i 1970'erne sammen med nye teknologiske produkter til afspilning af lyd og digitale kameraer til optagelse af billeder har revolutioneret vores anvendelse af digitale data i dagligdagen. Vi opbevarer og transformerer nu rutinemæssigt lydfiler og billedfiler i form af digitale data, som er nemme at udveksle elektronisk. Lyd- og billedfiler kan ikke blot komprimeres så de kun fylder en brøkdel af den oprindelige størrelse – de kan også bearbejdes på forskellig vis, så man kan trække detaljer frem eller reducere/fjerne fx uønsket støj. Alt dette er ikke bare muliggjort af teknologiske landvindinger indenfor fysikken, men også af væsentlige fremskridt indenfor den digitale såkaldte diskrete matematik.

Skal man pege på en enkelt afgørende landvinding indenfor den diskrete matematik, der har muliggjort den moderne digitale dataforarbejdning, er det især genopdagelsen i 1965 på IBM's forskningsinstitution Watson Research af en yderst effektiv algoritme til forarbejdning af digitale data, den såkaldte *Fast Fourier Transformation* med kælenavnet FFT. Det var den amerikanske statistiker James Cooley, der sammen med kollegaen John Tukey var hovedkræfterne bag dette. Algoritmen blev oprindeligt udarbejdet som et led i overvågningen af atomvåbensprængninger i det daværende Sovjetunionen. Med FFT til rådighed kunne amerikanerne langt hurtigere end tidligere bearbejde data fra seismiske sensorer i undergrunden i landene langs Sovjetunionen, og derved kunne de langt hurtigt fastlægge sprængningsstedet.

Du kan finde en video med en samtale mellem Cooley og Tukey her: https://ethw.org/Archives:History_of_FFT_with_Cooley_and_Tukey Samtalen giver et interessant indblik i forskningsmiljøet, hvordan forskere arbejder, og i en spændende historie om hvordan computere gav helt nye muligheder for den matematiske forskning



James Cooley (1926-2016)



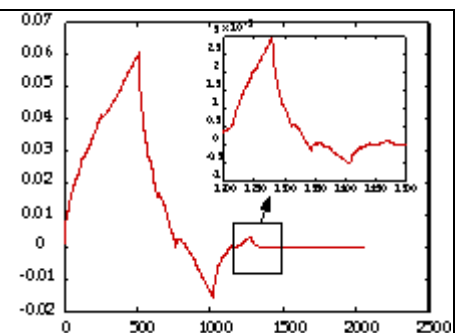
John Tukey (1915-2000)

Senere har man fundet andre effektive analysemetoder til bearbejdning af digitale data, ikke mindst den såkaldte *wavelet analyse*, der er en generalisering af fourier analysen. Den fik et kraftigt løft i slutningen af 80'erne, da den belgiske matematiker Ingrid Daubechies opdagede af en helt ny familie af bølgeformer, der er kontinuerte, men ikke differentiable, og som ikke kan fastlægges ved en simpel forskrift.

Bortset fra de allersimpleste Daubechies-funktioner er der nemlig tale om såkaldt fraktale funktioner, hvor grafen i det uendelige gentager sig selv på finere og finere skaler. Når man zoomer ind på et lille stykke af grafen dukker der altså et billede op som på mange måder ligner hele grafen.



Ingrid Daubechies der i 1988 overraskende fandt en helt ny serie af bølgeformer, der revolutionerede Wavelet analysen. Daubechies blev den første kvindelige professor i matematik ved Princeton Universitetet (1994-2010). Hun var præsident for det internationale matematikforbund IMU. i 2011-2014



Øvelse 1: Eksempel på en fraktal graf

- a) I dit værktøjsprogram er der en kommando til at udregne *brøkdelen af et helt tal*, fx $\text{mod}(x,1)$, der udregner resten ved division med 1. Benyt denne kommando til at definere en savtakfunktion:

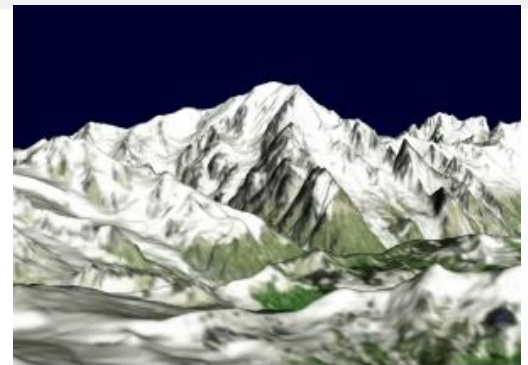
$$\text{tak}(x) := \frac{1}{2} - \left| \text{mod}(x,1) - \frac{1}{2} \right|. \text{ Tegn grafen for takfunktionen og kommentér dens udseende.}$$

- b) Vi kan *skalere* tak-funktionen, så den bliver mindre, idet vi hele tiden gør den en halv gang højere og en halv gang bredere: $\text{savtak}(x,k) := 2^{-k} \cdot \text{tak}(2^k \cdot x)$. Tegn de første 8 savtakfunktioner ($k = 0, 1, \dots, 7$) og kommentér deres udseende.

- c) Vi lægger nu successive savtakfunktioner sammen for at frembringe grafen for et fraktalt bjerg:

$$\text{sumsavtak}(x,n) := \sum_{k=0}^n \text{savtak}(x,k). \text{ Tegn grafen for de første 8 stadier i bjerget (n = 0, 1, \dots, 7) og kommentér deres udseende.}$$

- d) Vi definerer nu *Mont Blanc*-funktionen som den uendelige sum, dvs. $n = \infty$. I praksis må vi stoppe summen et passende højt sted afhængigt af computerens ydeevne. Tegn den følgende approksimation til Mont Blanc funktionen: $\text{sumsavtak}(x,127)$. Prøv derefter at zoome ind et passende sted på grafen, fx med en faktor 1000. Prøv derefter at zoome ind et passende sted på dit udsnit, fx med en faktor 1000.

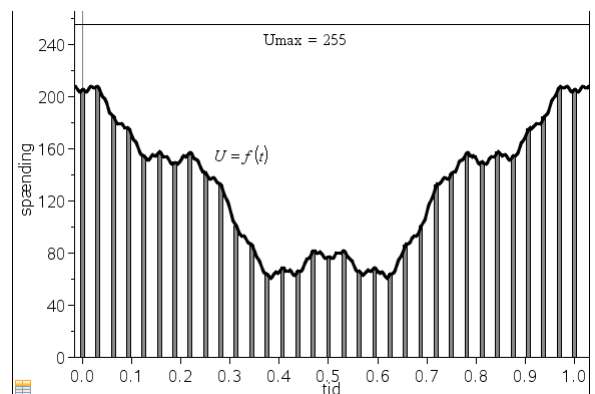


Mont Blanc – Europas højeste bjerg.

1. Digitale data

Vi starter med at se på hvordan man repræsenterer digitale data. Først ser vi på et lydsignal. Når vi optager en lyd med en mikrofon omsættes tryksvingningerne i mikrofonens membran til en hastig varierende spænding. Vi benytter nu en digitizer til at måle spændingen fx 44000 gange i sekundet og det er disse diskrete spændingsværdier $U_0, U_1, U_2, \dots, U_{44000}$ der udgør det digitale lydsignal. Fx kan en 8-bit digitizer måle $2^8 = 256$ forskellige spændingsniveauer. Disse spændingsniveauer kan fx repræsenteres ved tallene fra 0 til 255. Hvis fortegnet er vigtigt kan de også repræsenteres tallene $-127, \dots, -1, 0, 1, \dots, 127$, idet den første bit reserveres til fortegnet.

Ved lydoptagelser benyttes typisk en 16-bit digitizer, der derfor kan måle $2^{16} = 65536$ forskellige spændings- eller lyd-niveauer.



Spændingsfunktion $U = f(t)$, der samples 32 gange i sekundet. De *kontinuerte variable* t og U erstattes da af de *diskrete variable* $\text{tid} = \{0, 1/32, \dots, 31/32\}$ og $\text{spænding} = \{U_0, U_1, \dots, U_{31}\}$. Sammenhængen mellem de diskrete variable er vist som et søjlediagram.

I den digitale verden bruges ofte mål som kilobytes, Megabytes osv. I den ovenstående udregning

1 bit = 0 or 1
1 Byte = 8 bits

skal vi derfor bruge mere end 500 Megabytes. Hvis vi ikke vil komprimere dette datasæt skal vi derfor bruge fx en 650 MB cd-skive til at lagre musikken. Bruger vi en snedig komprimering af lydsignalet fx MP3-formatet fra 1993, der bygger på en variant af fourier-transformationen kan vi nøjes med meget mindre dataplads, typisk under 1/10 af det som det rå datasæt fyldte.

1 Byte dækker tallene fra 0 til 255 i binær notation

$$\text{Ex.: } 11001011_{\text{bin}} = 2^7 + 2^6 + 2^3 + 2^1 + 2^0 = 203_{\text{dec}}$$

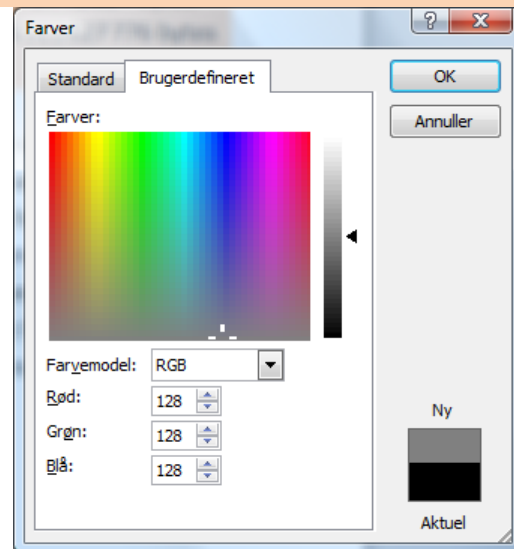
1 kilobyte = 1kB = 2^{10} bytes = 1024 bytes

1 Megabyte = 1MB = 2^{20} bytes = 1 048 675 bytes

1 Gigabyte = 1GB = 2^{30} bytes = 1 073 741 824 bytes

1 Terabyte = 1TB = 2^{40} bytes = 1 099 511 627 776 bytes

Hvis vi i stedet vil optage et gråtone billede (sort/hvid) med fx et digitalt kamera eller en digital scanner opdeles billedet i kvadratiske pixler (picture elements) og hver pixel tildeles en gråtoneværdi. Det digitale kamera indeholder en CCD-chip (Charged Coupled Device) opdelt i pixler, der registrerer hvor meget lys, der falder på hver pixel og omsætter det til en spænding, der kan måles med et digitalt voltmeter. CCD-chippen kan fx rumme 8 MegaPixler, dvs. 8 millioner pixler. I hver pixel registreres en gråtone som en byte, dvs. et niveau fra 0 til 255 med 255 = fuldt belyst = hvid og 0 = ingen lys = sort. Får vi brug for at farvebillede indskydes der farvefiltre, typisk med farverne rød, grøn og blå, og lysintensiteten registreres igen som en byte. Med 3 bytes til rådighed kan vi altså danne $2^8 \times 2^8 \times 2^8 = 2^{24} = 16\,777\,216$, dvs. over 16 millioner forskellige farver.



Farvekodning med RGB. Paletten er hentet fra Word. I den viste gråtoneskala er de tre farveværdier netop ens. Midtvejs fås altså en mellemgrå.

Øvelse 2 Fingeraftryk

I 1993 udviklede FBI en de facto standard for digitalisering og opbevaring af digitale fingeraftryk.

- FBI scanner fingeraftryk med en opløsning på 500 pixler pr tomme. Et enkelt fingeraftryk fylder 1.5×1.75 kvadrattommer. Der scannes i 8-bit gråtoner. Hvor meget fylder fingeraftrykkene fra to hænder?
- FBI har et arkiv med ca. 30 millioner menneskers fingeraftryk. Hvor meget fylder det, hvis det skal digitaliseres uden komprimering? En harddisk koster ca. 900 \$ pr GigaByte. Hvor meget koster det at digitalisere arkivet, hvis det sker uden komprimering.

Af grunde som vi vender tilbage til, er Fourier transformation ikke velegnet til at komprimere fingeraftryk. Så her anvender man i stedet den moderne wavelet transformation, der i langt højere grad er i stand til at bevare lokale detaljer.

I 1993 udviklede man også en generel standard JPEG for komprimering af digitale billeder (Joint Picture Experts Group). Den oprindelige standard var baseret på Fourier transformation, men i 2000 skiftede man over til en ny JPEG standard, der i stedet var baseret på wavelet teknikker udviklet af Daubechies.



Til venstre ses en original gråtonescanning af et fingeraftryk med 500 pixler pr tomme, til højre det samme digitale billede efter en wavelet-komprimering med faktoren 26.

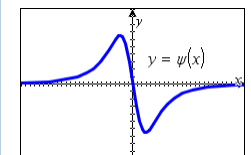
2. Wavelet analyse

Når man ser bølgeformer på havet, så er der tale om udsving fra et ligevægtsniveau, dvs. en blikstille hav-overflade. Bølgerne kan have mange former, men det er typisk for dem at de netop svinger lige meget op og ned.

Definition: Wavelet

En bølgeform eller en wavelet er en funktion $\psi(x)$, udtales psi-af-x, defineret på intervallet $[a; b]$, hvis graf ligger lige meget over som under x -aksen. Den afgrænser altså det samme areal over x -aksen som under x -aksen. En wavelet er derfor karakteriseret ved at såvel dens

integral som dens middelværdi er 0: $\int_a^b \psi(x) dx = 0$.



Øvelse 3: Konstruktion af wavelets

Det er særligt nemt at konstruere wavelets ud fra differentiation.

- Gør rede for at hvis den oprindelige funktion $f(x)$ har værdien 0 i grænserne, så vil den afledede funktion $\psi(x) = f'(x)$ netop være en wavelet-funktion.
- Det gælder også hvis definitionsmængden dækker over de reelle tal, hvis blot funktionen $f(x)$ går hurtigt mod nul i det uendelige: Et typisk eksempel er *Gaussfunktionen* $f(x) = e^{-\frac{x^2}{2}}$. Tegn graferne for Gaussfunktionen f og de to første afledede $\psi_1(x) = f'(x)$ og $\psi_2(x) = f''(x)$. Find også eksplicitte forskrifter for de to første afledede $\psi_1(x)$ og $\psi_2(x)$. Waveletten $\psi_2(x)$ knyttet til den anden afledede af Gauss-funktionen kaldes populært for Mexicanerhatten.

Wavelets findes i alle mulige former, men til wavelet analyse anvendes typisk *lokaliserede wavelets*, dvs. bølgeformer, der kun svinger i et meget kort tidsinterval. Udenfor dette interval, støtten for waveletten, er de altså nul. Andre wavelets som de i øvelsen konstruerede wavelets forsvinder aldrig helt, men de er i praksis alligevel lokaliserede, fordi de går så hurtigt mod nul.

I wavelet analyse anvender man en familie af wavelets, der alle stammer fra en grundform, en prototype, kaldet moder-waveletten. De andre wavelets i familien opstår så ved at man skalerer grundformen vandret og derefter og parallelforskyder den vandret. Derved opstår ved skaleringen en snævrere wavelet, som så ved vandret parallelforskydning kan bringes til at dække det samme tidsinterval som grundformen. Typisk skalerer man med faktorer, som er totalspotenser:

$$\psi_{jk}(x) = \psi(2^j \cdot x - k) = \psi\left(\frac{x - 2^{-j} \cdot k}{2^{-j}}\right)$$

Øvelse 4: Haar-waveletten

- Moderfunktionen* er defineret som den stykvis konstante funktion $\psi(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } 0 \leq x < \frac{1}{2} \\ -1 & \text{for } \frac{1}{2} \leq x < 1 \\ 0 & \text{for } 1 \leq x \end{cases}$.

Tegn grafen for moderfunktionen.

- Haar-waveletten er defineret som $\psi_{jk}(x) = \psi(2^j \cdot x - k)$, $0 \leq k < 2^j$.

Tegn graferne for de første tre zoomniveauer:

$$\psi_{10}(x), \psi_{11}(x)$$

$$\psi_{20}(x), \psi_{21}(x), \psi_{22}(x), \psi_{23}(x)$$

$$\psi_{30}(x), \psi_{31}(x), \psi_{32}(x), \psi_{33}(x), \psi_{34}(x), \psi_{35}(x), \psi_{36}(x), \psi_{37}(x)$$

og kommentér deres udseende.

Vi vil nu se nærmere på en bestemt særlig simpel form for wavelet analyse, der kaldes *Haar analyse*. Den har været kendt længe før betegnelsen wavelet analysen blev introduceret i 80'erne. Haar analysen er særlig simpel og de tilhørende wavelets indgår i mange wavelet familier. I Daubechies familie af grundlæggende waveletter er det grundformen D2.

Først vil vi imidlertid prøve at gøre os klart hvad vi ønsker at opnå med en funktionsanalyse som wavelet analyse eller fourier analyse. Man ønsker at transformere den oprindelige funktion over i en ny funktion, der dels viser karakteristiske træk ved den oprindelige funktion tydeligere, dvs. vi bruger den transformerede funktion til at trække informationer ud om den oprindelige funktion. Dels kan den transformerede funktion nemmere bruges til at trække ligegyldige detaljer ud af den oprindelige funktion, dvs. til at komprimere den oprindelige funktion, så vi ikke behøver lagre alle detaljerne, men kun de væsentlige detaljer.

Vi ser altså på en funktion f defineret på intervallet $[0;1[$ som vi digitaliserer ved at opdele intervallet $[0;1[$ i 8 lige store delintervaller med delepunkterne $x_0 = 0, x_1 = \frac{1}{8}, \dots, x_7 = \frac{7}{8}$. Funktionen er altså repræsenteret ved en liste over 8 datapunkter:

$$f = \{f(x_0), f(x_1), \dots, f(x_7)\} = \{y_0, y_1, y_2, y_3, y_4, y_5, y_6, y_7\}$$

Vi vil nu stille forskellige spørgsmål til egenskaberne ved funktionen. Langt det vigtigste er funktionens *niveau*, dvs. den gennemsnitsværdi funktionen har i intervallet. Den udregnes selvfølgelig som summen af funktionsværdierne divideret med antallet af datapunkter:

$$f_{\text{gennemsnit}} = \frac{f(x_1) + f(x_2) + \dots + f(x_8)}{8} = \frac{y_1 + y_2 + \dots + y_8}{8} = \frac{1}{8} \cdot y_1 + \frac{1}{8} \cdot y_2 + \dots + \frac{1}{8} \cdot y_8$$

I den sidste omskrivning har vi skrevet gennemsnittet som et vægtet gennemsnit, hvor alle vægtene er lige store. Vægtfunktionen er altså konstant og er givet ved listen

$$u_{00} = \left\{ \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8} \right\}$$

Vi kan så også skrive gennemsnittet som det vægtede gennemsnit:

$$f_{\text{gennemsnit}} = c_{00} = \text{sum}(u_{00} \cdot f)$$

Dette er den første wavelet koefficient c_{00} , og u_{00} er den første funktion i waveletanalysen, den såkaldte *faderfunktion*, der altså er konstant, og dermed *ikke* er en wavelet! Her er bogstavet c valgt for det engelske ord constant.

Det næste vi kunne interessere os for er så *forskellen på niveauet i den første halvdel og den sidste halvdel*, der også kan udregnes som et vægtet gennemsnit:

$$\begin{aligned} f_{\text{variation}} &= \frac{1}{2} \cdot (f_{\text{gennemsnit, første halvdel}} - f_{\text{gennemsnit, anden halvdel}}) = \frac{1}{2} \cdot \left(\frac{f(x_1) + \dots + f(x_4)}{4} - \frac{f(x_5) + \dots + f(x_8)}{4} \right) \\ &= \frac{1}{8} \cdot f(x_1) + \dots + \frac{1}{8} \cdot f(x_4) - \frac{1}{8} \cdot f(x_5) - \dots - \frac{1}{8} \cdot f(x_8) = \frac{1}{8} \cdot y_1 + \dots + \frac{1}{8} \cdot y_4 - \frac{1}{8} \cdot y_5 - \dots - \frac{1}{8} \cdot y_8 \end{aligned}$$

Denne gang er den første halvdel af vægtene altså positive og den anden halvdel negative. Vægtfunktionen er derfor denne gang givet ved

$$w_{00} = \left\{ \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, -\frac{1}{8}, -\frac{1}{8}, -\frac{1}{8}, -\frac{1}{8} \right\}$$

Vi kan så udregne variationen mellem første og anden halvdel som det vægtede gennemsnit:

$$f_{\text{variation}} = d_{00} = \text{sum}(w_{00} \cdot f)$$

Dette er den anden wavelet koefficient d_{00} og v_{00} den anden funktion i wavelet analysen, den såkaldte *moderfunktion*, der denne gang svinger symmetrisk op og ned, summen af vægtene er nul, og dermed er der netop tale om en wavelet! Her er bogstavet d valgt for det engelske ord detail.

Fortsætter vi på denne måde kan vi opdele intervallet i mindre og mindre dele og udregne de tilhørende wavelet-koefficienter, der fortæller noget om hvor hurtigt funktionen varierer på mindre og mindre delintervaller:

$$u_{00} = \left\{ +\frac{1}{8}, +\frac{1}{8}, +\frac{1}{8}, +\frac{1}{8}, +\frac{1}{8}, +\frac{1}{8}, +\frac{1}{8}, +\frac{1}{8} \right\} \quad c_{00} = \text{sum}(u_{00} \cdot f) = f_{\text{gennemsnit}} = \frac{y_0 + y_1 + y_2 + y_3 + y_4 + y_5 + y_6 + y_7}{8}$$

$$w_{00} = \left\{ +\frac{1}{8}, +\frac{1}{8}, +\frac{1}{8}, +\frac{1}{8}, -\frac{1}{8}, -\frac{1}{8}, -\frac{1}{8}, -\frac{1}{8} \right\} \quad d_{00} = \text{sum}(w_{00} \cdot f) = f_{\text{variation}} = \frac{1}{2} \cdot \left(\frac{y_0 + y_1 + y_2 + y_3}{4} - \frac{y_4 + y_5 + y_6 + y_7}{4} \right)$$

$$w_{10} = \left\{ +\frac{1}{4}, +\frac{1}{4}, -\frac{1}{4}, -\frac{1}{4}, 0, 0, 0, 0 \right\} \quad d_{10} = \text{sum}(w_{10} \cdot f) = f_{\text{variation over første halvdel}} = \frac{1}{2} \cdot \left(\frac{y_0 + y_1}{2} - \frac{y_2 + y_3}{2} \right)$$

$$w_{11} = \left\{ 0, 0, 0, 0, +\frac{1}{4}, +\frac{1}{4}, -\frac{1}{4}, -\frac{1}{4} \right\} \quad d_{11} = \text{sum}(w_{11} \cdot f) = f_{\text{variation over anden halvdel}} = \frac{1}{2} \cdot \left(\frac{y_4 + y_5}{2} - \frac{y_6 + y_7}{2} \right)$$

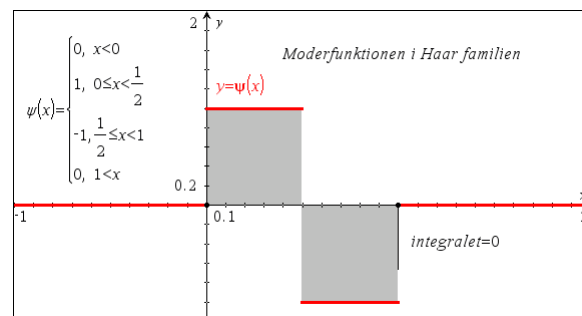
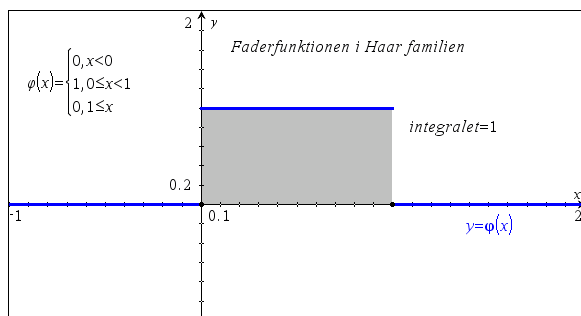
$$w_{20} = \left\{ +\frac{1}{2}, -\frac{1}{2}, 0, 0, 0, 0, 0, 0 \right\} \quad d_{20} = \text{sum}(w_{20} \cdot f) = f_{\text{variation over første fjerdedel}} = \frac{1}{2} \cdot (y_0 - y_1)$$

$$w_{21} = \left\{ 0, 0, +\frac{1}{2}, -\frac{1}{2}, 0, 0, 0, 0 \right\} \quad d_{21} = \text{sum}(w_{21} \cdot f) = f_{\text{variation over anden fjerdedel}} = \frac{1}{2} \cdot (y_2 - y_3)$$

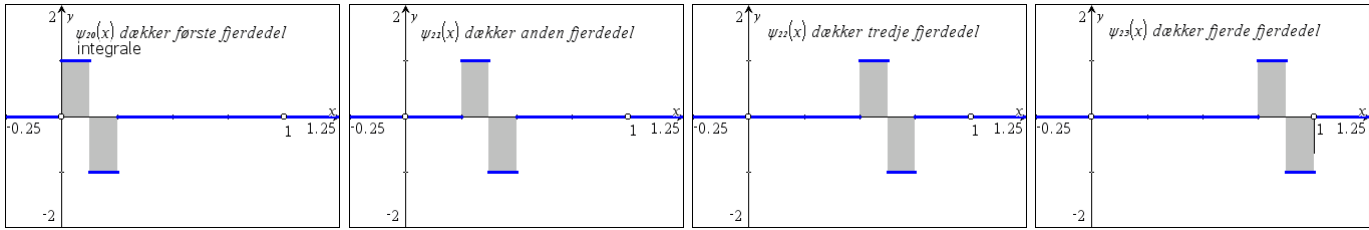
$$w_{22} = \left\{ 0, 0, 0, 0, +\frac{1}{2}, -\frac{1}{2}, 0, 0 \right\} \quad d_{22} = \text{sum}(w_{22} \cdot f) = f_{\text{variation over tredje fjerdedel}} = \frac{1}{2} \cdot (y_4 - y_5)$$

$$w_{23} = \left\{ 0, 0, 0, 0, 0, 0, +\frac{1}{2}, -\frac{1}{2} \right\} \quad d_{23} = \text{sum}(w_{23} \cdot f) = f_{\text{variation over fjerde fjerdedel}} = \frac{1}{2} \cdot (y_6 - y_7)$$

Alle de følgende vægtfunktioner svinger symmetrisk op og ned og er altså eksempler på wavelet-funktioner. De giver oplysninger om finere og finere detaljer i takt med at vi zoomer ind på mindre og mindre delintervaller. Både faderfunktionen $\phi(x)$ og moderfunktionen $\psi(x)$ kan opfattes som digitaliseringer af Haar-funktioner defineret for alle de reelle tal. Bemærk dig, at der ikke er tale om kontinuerte funktioner, idet de springer fra værdien 0 til værdien 1 og tilbage igen henholdsvis fra værdien 0 til værdien 1 til værdien -1 og tilbage igen til 0:



Ved vandrette skaleringer og forskydninger fås nu de øvrige wavelet-funktioner, her vist efter at vi har zoomet ind med zoomfaktoren 4:



Læg mærke til betegnelserne: $\psi_{j,k}(x)$ dækker det første indeks j over hvor mange gange vi har zoomet ind og det andet indeks k dækker over hvor mange gange vi har forskudt waveletten. De fire ovenstående Haar-waveletter hedder derfor $\psi_{20}(x)$, $\psi_{21}(x)$, $\psi_{22}(x)$ og $\psi_{23}(x)$, fordi vi har zoomet 2 gange ind (og dermed formindsket $2^2 = 4$ gange). Den første wavelet $\psi_{00}(x)$ er ikke forskudt, men starter i 0, den næste $\psi_{11}(x)$ er forskudt med én fjerdedel osv. I forhold til waveletterne $v_{j,k}$ mangler vi stadigvæk en lodret skalering. Men den er ret simpel idet Haar-waveletterne på zoom-niveau 0 skal ganges med $\frac{1}{8}$, på zoom-niveau 1 med $\frac{1}{4}$ og på zoom-niveau 2 med $\frac{1}{2}$.

Øvelse 5: Udregning af wavelet koefficienter

- a) Vi ser på det digitale signal $\text{signal} = \{1, 0, -3, 2, 1, 0, 1, 2\}$ defineret på intervallet $[0;1]$, dvs. signalet er optaget til tiderne $\{0, \frac{1}{8}, \frac{2}{8}, \frac{3}{8}, \frac{4}{8}, \frac{5}{8}, \frac{6}{8}, \frac{7}{8}\}$. Tegn grafen for dette signal. Hvad er dets mest karakteristiske træk? Hvor varierer det voldsomst?
- b) Indskriv vægtfunktionerne $u_{00}, w_{00}, w_{01}, w_{10}, w_{11}, w_{20}, w_{21}, w_{22}, w_{23}$ i dit værktøjsprogram. Udregn ved hjælp af formlerne:
 $c_{00} = \text{sum}(\text{signal} \cdot w_{00})$, $d_{00} = \text{sum}(\text{signal} \cdot w_{00})$, $d_{10} = \text{sum}(\text{signal} \cdot w_{10})$, ..., $d_{23} = \text{sum}(\text{signal} \cdot w_{23})$ de 8 wavelet-koefficienter, dvs. spektret $\text{spektrum} = \{c_{00}, d_{00}, d_{10}, d_{11}, d_{20}, d_{21}, d_{22}, d_{23}\}$ og tegn grafen for wavelet spektret som et søjlediagram med wavelet-koefficienterne op af andenaksen. Æn af wavelet-koefficienterne er uventet stor. Hvad fortæller den om signalet?

Nu er det ret simpelt at udregne wavelet-koefficienterne når vi arbejder med et digitalt signal med kun 8 værdier. Typiske digitale signaler vil have langt flere værdier, fx $2^{10} = 1024$ værdier. Så er det ikke længere hensigtsmæssigt at udføre alle kommandoerne $\text{sum}(\text{signal} \cdot w)$ i én lang rækkefølge. For meget omfattende udregninger kan beregningerne organiseres meget smartere, så man sparer værdifuld tid. Det er sådanne smarte algoritmer, der får tilnavnet fast! I Haar-analysen er der også en smart måde at organisere beregningerne på som vi viser i detalje med det samme eksempel som i øvelsen ovenfor.

Vi opskriver signalet i en tabel og opretter to søjler under signalet, en fadersøjle og en modersøjle. I fadersøjlen udregner vi systematisk gennemsnit, som den halve sum af to tal, i modersøjlen tilsvarende systematisk variationer, som den halve differens af to tal:

- a) Fadersøjlen udregnes ved at vi gennemløber signallisten *parvis* og udregner den halve sum af parrenen, dvs. deres gennemsnit. Det svarer til at finde niveauerne på zoomniveau 2, dvs. med en faktor 4, idet listen opbrydes i fire par. Derefter gennemløbes listen over gennemsnit parvis, og vi udregner deres gennemsnit. Det svarer til at finde niveauerne på zoomniveau 1, dvs. med en faktor 2. Endelig udregnes gennemsnittet af det sidst fundne par. Det svarer til niveauet på zoomniveau 0, dvs. hele signalfunktionens niveau eller wavelet-koefficienten c_{00} .

- b) Modersøjlen udregnes ved at vi gennemløber signallisten parvis og udregner den halve differens af parrene. Det svarer til at finde variationerne på zoomniveau 2, dvs. med en faktor 4, idet listen opbrydes i fire par. De fire fundne tal udgør altså wavelet-koefficienterne d_{20}, d_{21}, d_{22} og d_{23} . Derefter gennemløbes listen over niveauerne på zoomniveau 2 parvis, og vi udregner den halve differens, dvs. deres variation. Det svarer til at finde variationerne på zoomniveau 1, dvs. med en faktor 2. De to næste fundne tal udgør altså wavelet-koefficienterne d_{10} og d_{11} . Endelig udregnes den halve differens af det sidst fundne par. Det svarer til variationen på zoomniveau 0, dvs. hele signalfunktionens variation eller wavelet-koefficienten d_{00} .

I skemaform ser det således ud, hvor de gule rækker i modersøjlen er kopier af rækkerne i fadersøjlen:

	Fadersøjle: Halve summer Funktionens niveauer på delintervaller								Modersøjle: Halve differenser Funktionens variationer på delintervaller							
Zoomniveau 2	1	0	-3	2	1	0	1	2	1	0	-3	2	1	0	1	2
Zoomniveau 1	0.5		-0.5		0.5		1.5		$d_{20}=0.5$	$d_{21}=-2.5$	$d_{22}=0.5$	$d_{23}=-0.5$				
	0.5		-0.5		0.5		1.5		0.5	-0.5	0.5	1.5				
Zoomniveau 0	0.0				1.0				$d_{10}=0.5$				$d_{11}=-0.5$			
	0.0				1.0				0.0				1.0			
	$c_{00}=0.5$								$d_{00}=-0.5$							

Øvelse 6 Skema over fadersøjler og modersøjler

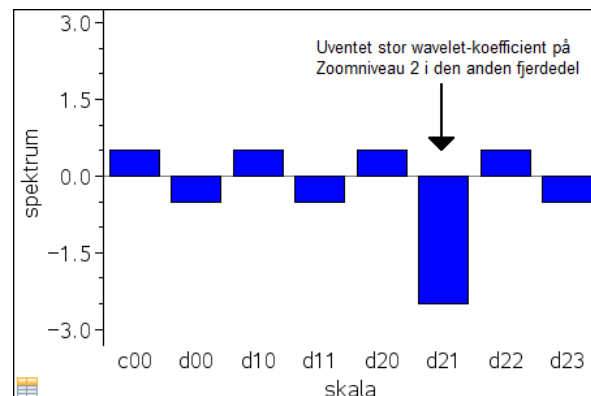
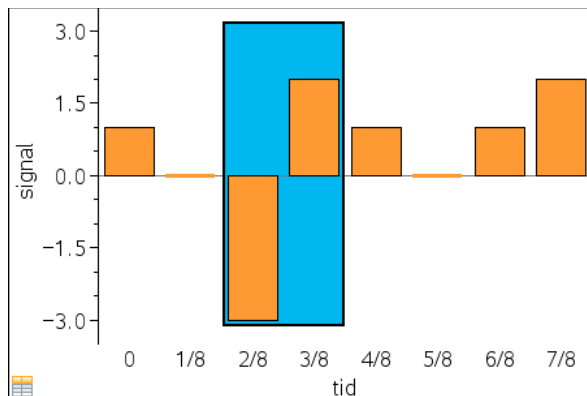
- a) Udfyld selv skemaet i hånden, idet du starter med at indsætte signalet i den øverste række og derefter arbejder ned gennem skemaet med halve summer i fadersøjlen og halve differenser i modersøjlen. Husk at kopiere de rækker du finder i fadersøjlen over i modersøjlen undervejs!
- b) Udfordring: Kan du udfylde skemaet baglæns, dvs. bruge skemaet til at finde signalet ud fra spektret?

	Fadersøjle: Halve summer Funktionens niveauer på delintervaller								Modersøjle: Halve differenser Funktionens variationer på delintervaller							
Zoomniveau 2																
Zoomniveau 1	c_{20}		c_{21}		c_{22}		c_{23}		$d_{20}=0.5$	$d_{21}=-2.5$	$d_{22}=0.5$	$d_{23}=-0.5$				
	c_{20}		c_{21}		c_{22}		c_{23}									
Zoomniveau 0	c_{10}				c_{11}				$d_{10}=0.5$				$d_{11}=-0.5$			
	c_{10}				c_{11}											
	$c_{00}=0.5$								$d_{00}=-0.5$							

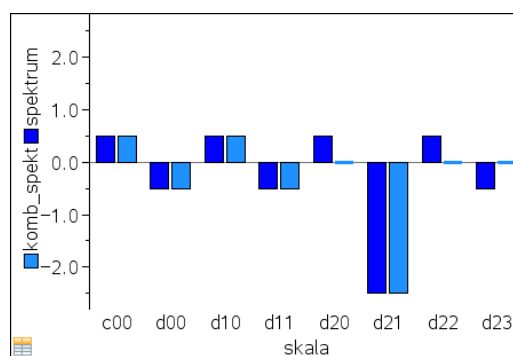
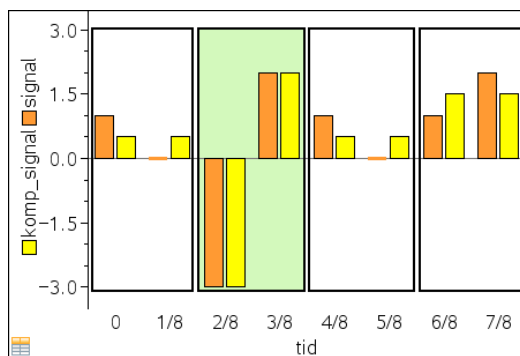
Vink: Hvordan finder man tilbage til to tal a og b , når man kender deres halve sum s og deres halve differens d ?

En af wavelet-koefficienterne $d_{21} = -2.5$ er uventet stor. Den peger netop på at der sker noget afgørende i det andet delinterval på zoomniveau 2 (der hvor vi opdeler i fjerdedele). Det kan vi også tydeligt se hvis vi

kigger på graferne for såvel signalet som spektret. Signalet varierer voldsomt, dvs. gør et stort spring, netop i den anden fjerdedel.



Hvis vi vil komprimere signalet, sker det typisk ved at smide detaljer ud som er ubetydelige, fx detaljerne fra det højeste zoomniveau. Men i dette tilfælde skal vi altså beholde d_{21} og nøjes med at sætte d_{20} , d_{22} og d_{23} til 0. Hvis man rekonstruerer det signal, der hører til det komprimerede spektrum fås da:



I den anden fjerdedel bevarer vi det fulde udsving, men i de øvrige fjerdele udjævner vi signalet og erstat- ter det med gennemsnitsværdien. Da signalet kun varierer langsomt i de øvrige fjerdedele følger det kom- primerede signal derfor det oprindelige signal rimelig tæt.

Alt dette er selvfølgelig noget grovkornet, fordi vi kun arbejde med en digitalisering i 8 punkter. Det bliver først interessant, når man kan arbejde med digitaliseringer med tusindvis af punkter, fx $2^{12} = 4096$ punk- ter, hvilket giver mulighed for 12 zoom-niveauer. Så er det håbløst at udføre wavelet-analysen i hånden. Men selv på computer tager det langt tid og derfor er den viste hurtige algoritme til at udføre Haar-analy- sen ved hjælp af successive halve summer og differenser af stor betydning for den praktiske anvendelse af Haar-analyse. Det er også her at komprimeringen kommer til sin fulde ret. Om vi kan smide 3 wavelet- komponenter væk i en 8 punkts analyse spiller ikke den store rolle, da analysen under alle omstændighe- der går stærkt. Men kan vi tilsvarende smide over 2000 wavelet-komponenter væk i en 4096 punkts ana- lyse, begynder det at betyde noget og kan vi nøjes med de 10 laveste niveauer er der tale om at smide over 3000 wavelet-komponenter væk og så begynder det at batte. Moderne wavelet-analyser af fx finger- aftryk bygger på en komprimering med en faktor over 20 uden at man kan se forskel på original og kopi.

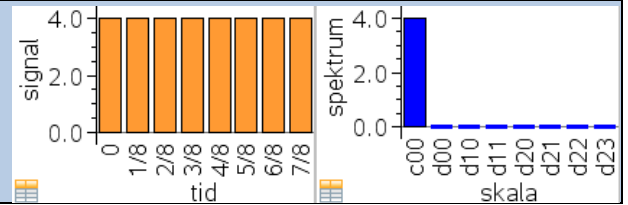
Bemærkning: Det viser sig dog at være en svaghed ved Haar-analysen at den bygger på diskontinuerte wavelets og derfor støtter man sig i dag på de mere raffinerede wavelets fundet af Daubechies.

3. Den inverse wavelet transformation

Selv om der som øvelsen viser findes en hurtig algoritme til at rekonstruere signalet ud fra spektret kan det betale sig at forstå den langsomme algoritme, da den viser væsentlige egenskaber ved signalanalysen. Vi starter med en simpel men vigtig observation:

Sætning: Waveletspektret for et konstant signal

Hvis signalet s er konstant, dvs. $s = k$, består wavelet-spektrum kun af konstantleddet $c_{00} = k$.



Beviset er meget simpelt: Konstantleddet $c_{00} = \text{sum}(s \cdot u_{00}) = k \cdot \text{sum}(u_{00}) = k$, idet faderfunktionen u_{00} er konstant og netop giver gennemsnittet af signalet. For de øvrige wavelet-koefficienter fås $d_{jk} = \text{sum}(s \cdot w_{jk}) = k \cdot \text{sum}(w_{jk}) = 0$, idet wavelet-funktionen w_{jk} jo netop er en wavelet, hvor summen af værdierne er 0, idet der ligger lige meget over og under akse.

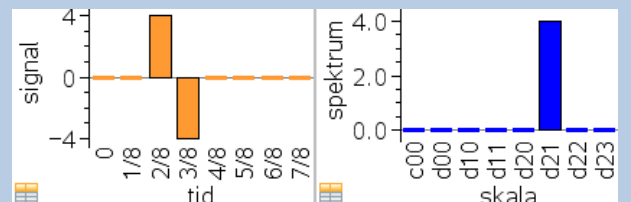
For ethvert andet signal kan vi trække niveauet ud af signalet, dvs. skrive signalet på formen **signal** = c_{00} + rest, hvor resten må selv være en wavelet, dvs. have middelværdi 0. Det er derfor nærliggende at forsøge at skrive resten som en superposition af wavelets, dvs. skrive den på formen:

$$\text{signal} = c_{00} + d_{00}^* \cdot w_{00} + d_{10}^* \cdot w_{10} + d_{11}^* \cdot w_{11} + d_{20}^* \cdot w_{20} + d_{21}^* \cdot w_{21} + d_{22}^* \cdot w_{22} + d_{23}^* \cdot w_{23}$$

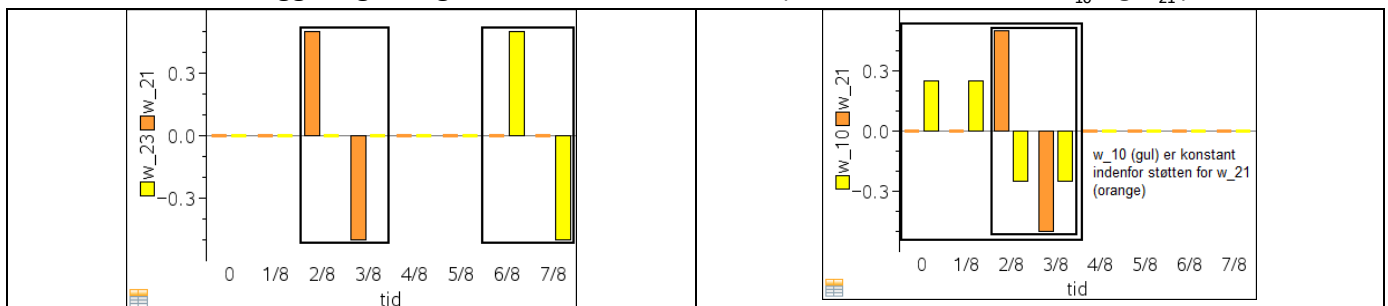
Vi skal lige om lidt se at koefficienterne er tæt knyttet til wavelet-koefficienterne. Men først viser vi den følgende udvidelse af sætningen om spektret for et konstant signal:

Sætning: Wavelet-spektrum for en Haar-wavelet.

Hvis signalet er givet ved en Haar-wavelet $\psi_{jk}(x)$ består spektret kun af wavelet-koefficienten d_{jk} hørende til $\psi_{jk}(x)$.



Beviset er igen meget simpelt: Faderkoefficienten c_{00} forsvinder fordi middelværdien af en enhver Wavelet er 0. For moderkoefficienterne fås: $c_{nm} = \text{sum}(\text{signal} \cdot w_{nm}) = k \cdot \text{sum}(w_{jk} \cdot w_{nm})$, idet Haar-waveletten $\psi_{jk}(x)$ er proportional med w_{jk} . Men hvis w_{jk} og w_{nm} hører til samme Zoomniveau, dvs. $j = n$ (men $k \neq m$), så har de støtter der ikke overlapper, dvs. en af faktorerne er nødvendigvis nul (se illustrationen for w_{21} og w_{23}). Hvis de hører til forskellige zoomniveauer, dvs. $j \neq n$, så vil waveletten med det laveste zoomniveau være konstant – eventuelt 0 – hen over støtten for den anden wavelet og summen vil derfor forsvinde, fordi den anden wavelet ligger lige meget over som under akse (se illustrationen for w_{10} og w_{21}):



Hvis et signal er en superposition af Haar-wavelets

$$\text{signal} = c_{00} + d_{00}^* \cdot w_{00} + d_{10}^* \cdot w_{10} + d_{11}^* \cdot w_{11} + d_{20}^* \cdot w_{20} + d_{21}^* \cdot w_{21} + d_{22}^* \cdot w_{22} + d_{23}^* \cdot w_{23}$$

kan vi nu nemt finde sammenhængen mellem koefficienterne. Fx fås

$$d_{21}^* = \text{sum}(\text{signal} \cdot w_{21}) = \text{sum}(d_{21}^* \cdot w_{21} \cdot w_{21}) = d_{21}^* \cdot \text{sum}(w_{21}^2) = d_{21}^* \cdot \frac{1}{2}$$

fordi alle de andre led forsvinder ifølge den foregående sætning og fordi $w_{21}^2 = \{0, 0, \frac{1}{4}, \frac{1}{4}, 0, 0, 0, 0\}$. Vi ser altså at der gælder den overraskende simple sammenhæng: $d_{21}^* = 2 \cdot d_{21}$.

Tilsvarende sammenhænge fås for alle de andre koefficienter. Men *ethvert* signal kan skrives som en superposition af Haar-wavelets. Hvis vi fx digitaliserer med 8 punkter er der 8 signalværdier og 8 koefficienter, så superpositionen svarer til 8 lineære ligninger med 8 ubekendte. Selv om der findes specielle omstændigheder hvor et sådant ligningssystem ikke kan løses er Haar-waveletten netop valgt med omhu, så ligningssystemet kan løses. Vi finder derfor:

Sætning: Den inverse wavelet-transformation

For en 8-punkts wavelet transformation med spektrum $\{c_{00}, d_{00}, d_{10}, d_{11}, d_{20}, d_{21}, d_{22}, d_{23}\}$ er signalet givet ved den følgende superposition af Haar-wavelets:

$$\begin{aligned} \text{signal} &= c_{00} + 8d_{00} \cdot w_{00} + 4 \cdot (d_{10} \cdot w_{10} + d_{11} \cdot w_{11}) + 2 \cdot (d_{20} \cdot w_{20} + d_{21} \cdot w_{21} + d_{22} \cdot w_{22} + d_{23} \cdot w_{23}) \\ &= 8 \cdot (c_{00} \cdot u_{00} + d_{00} \cdot w_{00}) + 4 \cdot (d_{10} \cdot w_{10} + d_{11} \cdot w_{11}) + 2 \cdot (d_{20} \cdot w_{20} + d_{21} \cdot w_{21} + d_{22} \cdot w_{22} + d_{23} \cdot w_{23}) \\ &= c_{00} \cdot \phi + d_{00} \cdot \psi + d_{10} \cdot \psi_{10} + d_{11} \cdot \psi_{11} + d_{20} \cdot \psi_{20} + d_{21} \cdot \psi_{21} + d_{22} \cdot \psi_{22} + d_{23} \cdot \psi_{23} \end{aligned}$$

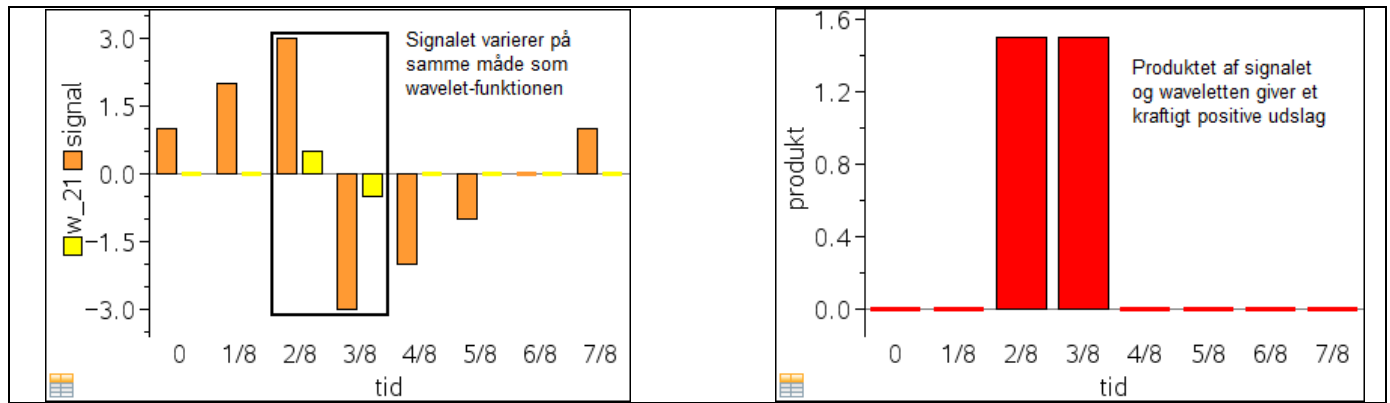
Sætningen generaliseres uden videre til wavelet-analyser med tusindvis af datapunkter – men så skal det under alle omstændigheder flyttes ind på computeren.

Øvelse 7: Wavelet-analyser

- Gennemfør wavelet-analysen med signalet $\text{signal} = \{1, 0, -3, 2, 1, 0, 1, 2\}$, hvor du komprimerer signalet ved at fjerne leddene med wavelet-funktionerne $\psi_{20}(x), \psi_{22}(x), \psi_{23}(x)$ men altså *ikke* leddet med $\psi_{21}(x)$, og derefter rekonstruerer det komprimerede signal.
- Gennemfør en wavelet-analyse af signalet $\text{signal} = \{2, 2, 1, 1, -1, -1, -2, -2\}$. Kommentér spektret!
- Gennemfør en wavelet-analyse af signalet $\text{signal} = \{-3, -1, 1, 3, 3, 1, -1, -3\}$. Kommentér spektret!

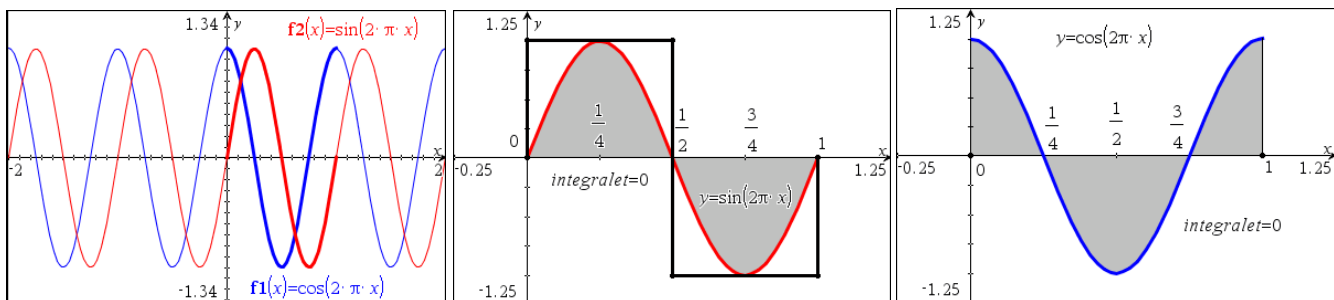
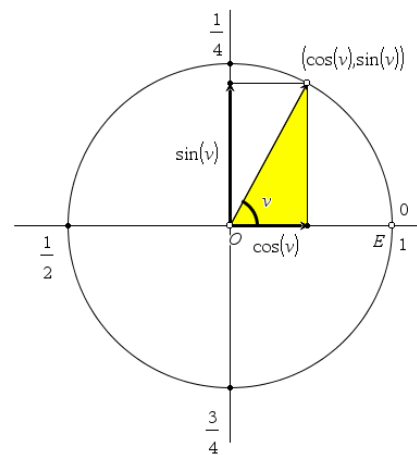
Vi ser nu at signal-analysen består i en *udfoldning* af de wavelet-funktioner, der ligger skjult inde i signalet. For at finde ud af om der fx ligger en skjult wavelet-funktion $\psi_{21}(x)$ ganger vi signalet med waveletten: $s(x) \cdot \psi_{21}(x)$. Hvis *signalet varierer på samme måde som waveletten*, vil produktet give store positive værdier, fordi signalet vil have tendens til at være positiv samtidigt med waveletten og negativ samtidigt med waveletten.

Signalet kan også variere lige modsat waveletten, hvorfor produktet vil give store negative værdier, fordi signalet vil have tendens til at være positiv samtidigt med at waveletten er negativ og negativ samtidigt med at waveletten er positiv. Men hvis fx signalet er konstant hen over wavelettens støtte giver det vægtede gennemsnit nul, og signalet indeholder derfor *ikke* en skjult wavelet-funktion $\psi_{21}(x)$.



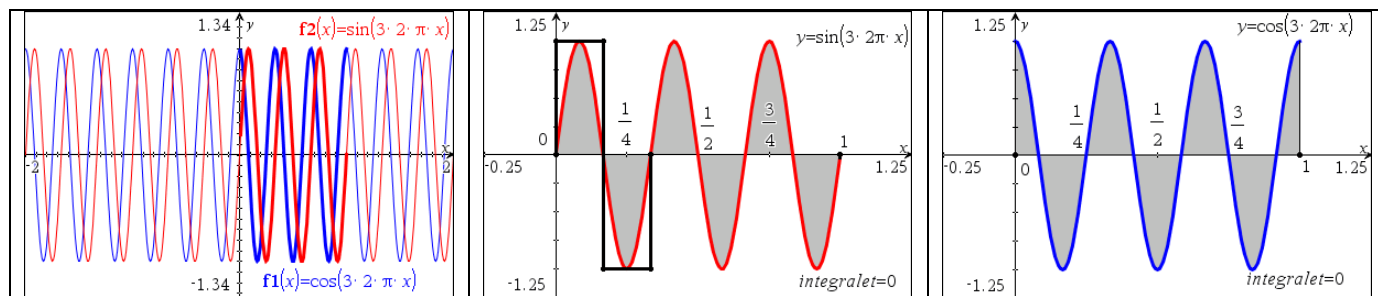
4. Fourier analyse

Vi vender os så mod Fourier analysen, der beror på et særligt simpelt og heldigt valg af wavelet-funktioner i form af de harmoniske svingninger, der kan beskrives enten ved cosinus- eller sinus-funktioner. De trigonometriske funktioner er tæt knyttet til enhedscirklen. For bedre at kunne sammenholde dem med fx Haar wavelets vil vi benytte *vindingstallet* som vinkelmål, dvs. vi måler en vinkel som den brøkdelen, den udgør af hele cirkelens omkreds. Da de trigonometriske funktioner traditionelt regnes i radianer, hvor vinklen måles som buelængden på en enhedscirkel svarer vindingstallet x til radiantallet $2\pi \cdot x$. Hvis vi måler vinklen i grader svarer vindingstallet i stedet til $360^\circ \cdot x$. Graferne for de grundlæggende trigonometriske funktioner sinus og cosinus ser da således ud:



Denne gang er waveletten *ikke* lokaliseret. I stedet er grundsvingningerne periodiske og svinger symmetrisk op og ned omkring x -aksen. Middelværdien over en periode er derfor stadigvæk 0, som det sig hør og bør for en wavelet. Men hvis vi indskrænker os til at se på periodeintervallet $[0;1]$, kan man godt se at sinussvingningen er en blød udgave af Haar-waveletten.

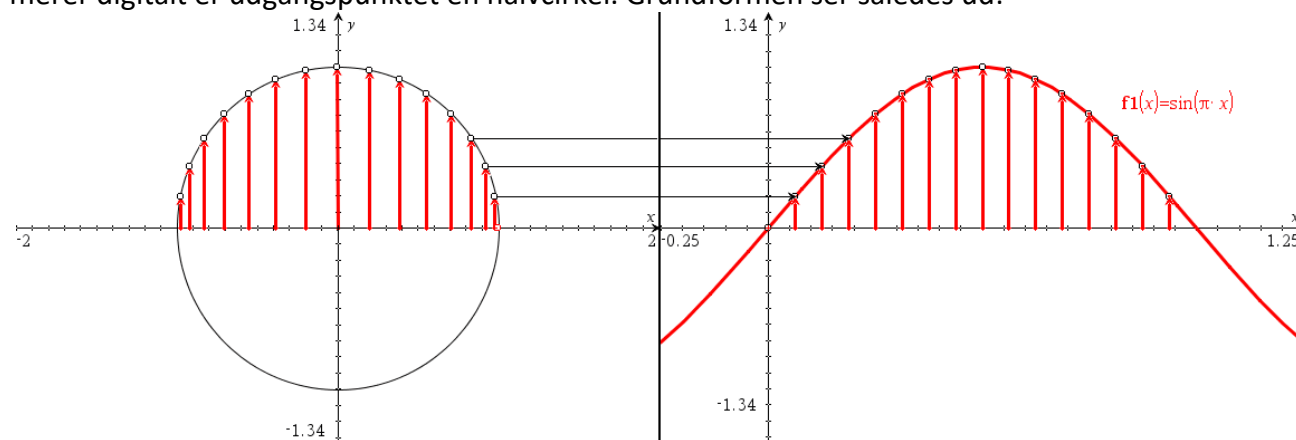
Ligesom i de andre waveletfamilier skalerer man grundsvingningerne, svarende til grundtonerne, og indfører de afledte svingninger, *overtonerne* med *højere frekvenser*, dvs. flere bølger pr tidsenhed: $\cos(k \cdot 2\pi \cdot x)$ og $\sin(k \cdot 2\pi \cdot x)$, $k = 2, 3, 4, 5, \dots$. Det svarer til at vi i stedet for én gang, nu drejer k gange rundt på enhedscirklen:



Men i modsætning til de andre wavelet-familier forskyder man *ikke* de afledte svingninger. Ovenfor har vi fx vist de trigonometriske waveletter svarende til $k = 3$. Hvis vi fx forskød sinus-svingningen ville den bare gå over i sig selv. De trigonometriske waveletter er periodiske og giver netop ikke mulighed for lokaliserer i tid. Af samme grund bruger man heller ikke så meget skala-begrebet i forbindelse med fourier-analysen. Nok zoomer man ind så man kan se finere detaljer, men man kan ikke se, hvor detaljerne befinder sig på tidsaksen. De er smurt ud i tid. I stedet hæfter man sig ved frekvensen af svingningen. Så detaljer, der hører til en bestemt frekvens dukker op i frekvens-spektret ved den pågældende frekvens.

5. Digital Fourier transformation

Vi går nu over til at se på digitale signaler, hvor vi altså kun måler signalet til bestemte tidspunkter. Hvis vi igen måler med 8 punkter i et tidsinterval svarer det til at opdele enhedsintervallet i 8 lige store delintervaller og se på værdierne for de trigonometriske waveletter i delepunkterne. Men når vi fourier transformerer digitalt er udgangspunktet en halvcirkel. Grundformen ser således ud:



Her ses en udfoldning af en 16-punkt sinus-wavelet ud fra de øverste hjørner i en regulære 32-kant. Den diskrete graf gengiver rimeligt godt den kontinuerte grafs forløb. Men det er klart at jo færre punkter vi vælger, jo ringere bliver ligheden. Læg mærke til at grundformen *ikke* er en wavelet, da den jo kun udgør en halv bølgelængde! Den fungerer i stedet som en form for faderfunktion. Læg også mærke til at da både faderfunktionen $\sin(\pi \cdot x)$ og moderfunktionen $\sin(2\pi \cdot x)$ forsvinder i $x = 0$ kommer det samme til at gælde alle sinus-waveletterne og en diskret sinustransformation kan derfor *kun* anvendes på signaler, der forsvinder til tiden $t = 0$. I praksis anvender man derfor mest den diskrete cosinustransformation, men da den er lidt mere teknisk at forklare, holder vi os her til sinustransformationerne. For at holde regneriet nede og gøre det nemmere at overskue ser vi også kun på den diskrete 8-punkt sinus-transformation.

Øvelse 8: Sinus wavelets

- a) Opret lister over vægtene hørende til $u_1 = \left\{ \frac{1}{2} \cdot \sin(\pi \cdot \frac{0}{8}), \frac{1}{2} \cdot \sin(\pi \cdot \frac{1}{8}), \frac{1}{2} \cdot \sin(\pi \cdot \frac{2}{8}), \dots, \frac{1}{2} \cdot \sin(\pi \cdot \frac{7}{8}) \right\}$ og sinus-waveletterne w_2, w_3, \dots, w_7 , dvs. $w_2 = \left\{ \frac{1}{2} \cdot \sin(2 \cdot \pi \cdot \frac{0}{8}), \frac{1}{2} \cdot \sin(2 \cdot \pi \cdot \frac{1}{8}), \frac{1}{2} \cdot \sin(2 \cdot \pi \cdot \frac{2}{8}), \dots, \frac{1}{2} \cdot \sin(2 \cdot \pi \cdot \frac{7}{8}) \right\}$
 $w_3 = \left\{ \frac{1}{2} \cdot \sin(3 \cdot 2\pi \cdot \frac{0}{8}), \frac{1}{2} \cdot \sin(3 \cdot 2\pi \cdot \frac{1}{8}), \frac{1}{2} \cdot \sin(3 \cdot 2\pi \cdot \frac{2}{8}), \dots, \frac{1}{2} \cdot \sin(3 \cdot 2\pi \cdot \frac{7}{8}) \right\}$ osv.
- b) Tegn graferne for sinus-waveletterne. Kontrollér at summerne af værdierne for sinus-waveletterne er 0.

Når vi skal Fourier-analysere et signal s går vi nu frem på samme måde som ved Haar-analysen. Vi illustrerer det her med en diskret sinus-transformation:

Øvelse 9: Diskret sinus-transformation

- a) Opret lister over vægtene hørende til faderfunktionen u_1 og sinus-waveletterne w_2, w_3, \dots, w_7 , jfr. øvelse 8. Som signal bruger vi savtak-waveletten **signal** = $\{0, 1, 2, 1, 0, -1, -2, -1\}$. Udregn Fourier-koefficienterne som de vægtede gennemsnit, dvs. $b_1 = \text{sum}(\text{signal} \cdot u_1)$ og $b_2 = \text{sum}(\text{signal} \cdot w_2), \dots, b_7 = \text{sum}(\text{signal} \cdot w_7)$.
- b) Opret grafer for signalet som funktion af tiden og for Fourierspektret, dvs. Fourier-koefficienterne som funktion af frekvensen. Kommentér graferne.
- c) Prøv derefter at Fourier-analysere sinus-waveletten svarende til en ren svingning med frekvensen 3. $w_3 = \left\{ \frac{1}{2} \cdot \sin(3 \cdot 2\pi \cdot \frac{0}{8}), \frac{1}{2} \cdot \sin(3 \cdot 2\pi \cdot \frac{1}{8}), \frac{1}{2} \cdot \sin(3 \cdot 2\pi \cdot \frac{2}{8}), \dots, \frac{1}{2} \cdot \sin(3 \cdot 2\pi \cdot \frac{7}{8}) \right\}$. Konklusion?

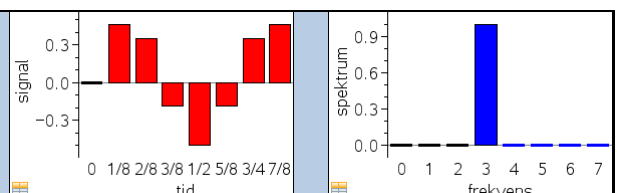
Det er nu nærliggende at forsøge at skrive signalet som en superposition af sinus-wavelets, dvs. skrive den på formen:

$$\text{signal} = b_1^* \cdot u_1 + b_2^* \cdot w_2 + b_3^* \cdot w_3 + b_4^* \cdot w_4 + b_5^* \cdot w_5 + b_6^* \cdot w_6 + b_7^* \cdot w_7$$

Vi skal lige om lidt se at koefficienterne er identiske med wavelet-koefficienterne. Men først nævner vi den fundamentale sætningen om spektret for et signal, der er en ren sinus wavelet:

Sætning: Fourierspektret for en sinus-wavelet.

Hvis signalet er givet ved en sinus-wavelet $\psi_k(x) = \sin(k \cdot \pi \cdot x)$ består Fourierspektret kun af Fourier-koefficienten b_k hørende til $\psi_k(x)$.



Vi vil ikke her bevise sætningen, der kræver anvendelse af en række trigonometriske identiteter. Men ideen er at de vægtede gennemsnit $\text{sum}(w_k \cdot w_m)$ automatisk forsvinder, hvis frekvenserne er forskellige, dvs. $k \neq m$. Hvis derimod frekvenserne er de samme fås:

$$\text{sum}(w_1^2) = \text{sum}(w_2^2) = \text{sum}(w_3^2) = \text{sum}(w_4^2) = \text{sum}(w_5^2) = \text{sum}(w_6^2) = \text{sum}(w_7^2) = 1$$

Hvis et signal er en superposition af sinus-bølger

$$\text{signal} = b_1^* \cdot u_1 + b_2^* \cdot w_2 + b_3^* \cdot w_3 + b_4^* \cdot w_4 + b_5^* \cdot w_5 + b_6^* \cdot w_6 + b_7^* \cdot w_7$$

kan vi nu nemt finde sammenhængen mellem koefficienterne. Fx fås

$$b_3 = \text{sum}(\text{signal} \cdot w_3) = \text{sum}(b_3^* \cdot w_3 \cdot w_3) = b_3^* \cdot \text{sum}(w_3^2) = b_3^*$$

fordi alle de andre led forsvinder ifølge den foregående sætning og fordi $\sum(w_3^2)=1$. Vi ser altså at der gælder den overraskende simple sammenhæng: $b_3^* = b_3$. Tilsvarende sammenhænge fås for alle de andre fourier-koefficienter. Vi samler det i sætningen

Sætning: Den inverse sinus-transformation

For en 8-punkts diskret cosinus transformation med spektrum $\{0, b_1, b_2, b_3, b_4, b_5, b_6, b_7\}$ er signalet givet ved den følgende superposition af sinus-wavelets:

$$\text{signal} = b_1 \cdot u_1 + b_2 \cdot w_2 + a_3 \cdot w_3 + a_4 \cdot w_4 + a_5 \cdot w_5 + a_6 \cdot w_6 + a_7 \cdot w_7$$

Sætningen generaliseres uden videre til fourier-analyser med tusindvis af datapunkter – men så skal det under alle omstændigheder flyttes ind på computeren. Samtidigt skal vi finde en hurtigere metode til at beregne spektret. Med tusindvis af datapunkter tager det alt for lang tid bare at gange vægte på i en uendelighed.

Der findes imidlertid netop et trick, FFT, der gør det meget hurtigere at udregne Fourier koefficienterne. Vi kan ikke her give alle detaljerne, men ideen er den at vi splitter signalet i to delsignaler: Et indeholdende de lige koefficienter og ét indeholdende de ulige koefficienter. Hvis fx der er tale om et 16-punkt signal kommer vægtene derfor fra de øverste 16 vinkelspidser i en regulær 32-kant indskrevet i enhedscirklen.

De lige koefficienter kommer da fra de 8 vinkelspidser i en regulær 16-kant, der er indskrevet på samme måde i enhedscirklen. Men de ulige koefficienter kommer desværre fra en regulære 16-kant, der er drejet i forhold til den foregående 16-kant. Ved hjælp af i princippet simple trigonometriske identiteter kan vi dreje den ulige 16-kant, så den stemmer overens med den lige 16-kant.

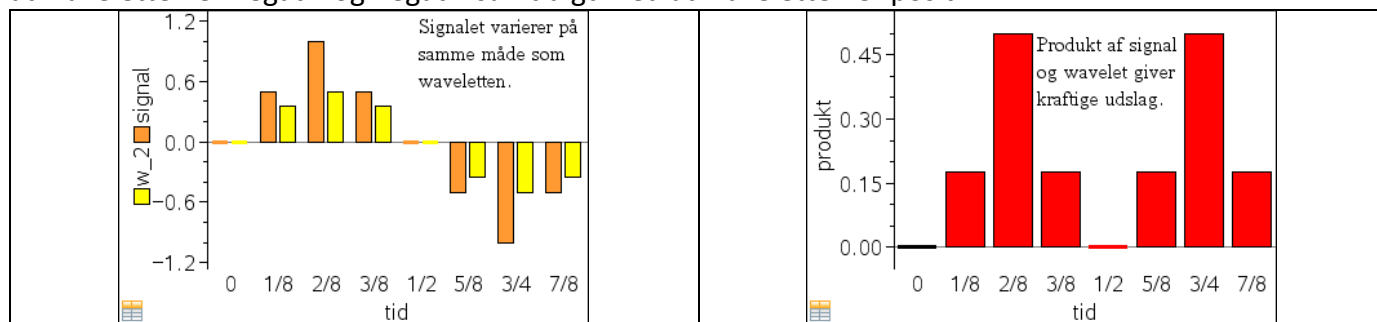
Det samlede vægtgennemsnit er da splittet som en sum af et bidrag fra de lige led og et bidrag fra de ulige led. Bidraget fra de lige led, stammer netop fra et vægtet gennemsnit baseret på et 8-punktsignal og bidraget fra de ulige led kan på forholdsvis simpel vis drejes, så også de stammer fra et vægtet gennemsnit baseret på et 8-punktsignal. Sådan fortsætter vi med at splitte delsignalerne indtil udregningen er trivial. Dette er hovedideen bag FFT.

Øvelse 10: Fourieranalyse af simple signaler

- Gennemfør fourier-analysen med firkantsignalet $\text{signal} = \{0, 0, 4, 4, -4, -4, 0, 0\}$, hvor du komprimerer signalet ved at fjerne leddene med de to højeste frekvenser, og derefter rekonstruerer det komprimerede signal.
- Gennemfør en fourier-analyse af signalet $\text{signal} = \{0, 1, 2, 3, 4, 3, 2, 1\}$. Kommentér spektret!
- Gennemfør en fourier-analyse af signalet $\text{signal} = \{0, 1, 0, -1, 0, 1, 0, -1\}$. Kommentér spektret!

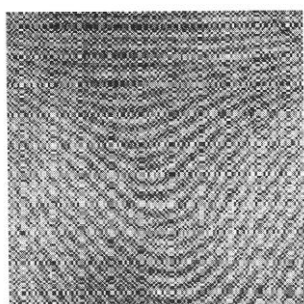
Vi ser nu at Fourier-analysen består i en *udfoldning* af de sinusbølge-funktioner, der ligger skjult inde i signalet. For at finde ud af om der fx ligger en skjult sinus-bølge $\sin(2 \cdot \pi \cdot x)$ inde i signalet ganger vi signalet med sinus-waveletten: $s(x) \cdot \sin(2 \cdot \pi \cdot x)$. Hvis *signalet varierer på samme måde som waveletten*, vil produktet give store positive værdier, fordi signalet vil have tendens til at være positiv samtidigt med waveletten og negativ samtidigt med waveletten. Signalet kan også variere lige modsat waveletten, hvorfor

produktet vil give store negative værdier, fordi signalet vil have tendens til at være positiv samtidigt med at waveletten er negativ og negativ samtidigt med at waveletten er positiv.

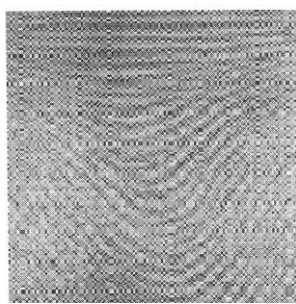


Den diskrete Fourieranalyse som vi her har præsenteret den går i det væsentlige tilbage til Gauss i 1805, hvor han noterer sig at 'Erfaringen vil lære brugeren, at denne metode vil lette byrden i de mekaniske beregninger betydeligt.' Gauss selv gjorde dog ikke meget reklame for den diskrete Fourieranalyse og overlod derfor for en gangs skyld æren til andre, ikke mindst Fourier, der i 1807 fandt den kontinuerte version af Fourier-analysen. Fouriers arbejde blev i starten mødt med betydelig skepsis. Men siden hen gik metoden sin sejrsgang og blev en hjørnesten i udviklingen af den moderne matematik. I forbindelse med den digitale revolution og genopdagelsen af FFT-algoritmen i 1965, blev Fourier-analysen også den helt centrale metode i analysen af diskrete signaler. Det var helt afgørende at man pludselig fik stillet en metode til rådighed, der var skræddersyet til forarbejdning af store datamængder.

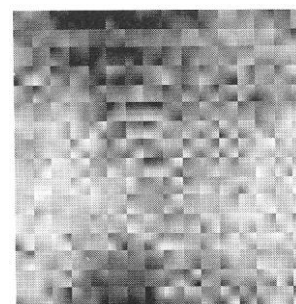
Metoden er dog ikke helt problemfri. Selv ved analyse af lydsignaler rummer der fra starten et klart problem: Hvis vi fx ser på to lydsignaler, hvoraf det første består af en superposition af fx fire rene toner overlejret med lidt støj og det andet består af de samme fire rene toner spille efter tur meget hurtigt efter hinanden og igen overlejret med lidt støj, vil fourier-spektrene i praksis være ens: Spektrene vil vise, at der i lydsignalet findes fire meget klare frekvenser, men de vil ikke kunne skelne mellem om de fire lyde optræder samtidigt eller hver for sig. Hvis man i dag vil rense gamle originale lydoptagelser foretrækker man derfor at anvende moderne wavelet-analyser, hvor man i langt højere grad kan skelne forskellige detaljer fra hinanden.



Fingeraftryk original
128×128 pixels



Wavelet-kompression baseret på
ca. 1/5 af koefficienterne



Fourier-kompression baseret på
ca. 1/5 af koefficienterne.

Kilde. *Mathematics of digital images: Creation, Compression, Restoration, Recognition of Hoggar, Cambridge*

Noget tilsvarende gør sig gældende i billedanalysen. Da fx FBI i 90'erne skulle vælge en kompressionsmetode til lagring af digitale fingeraftryk undersøgte de også muligheden for at anvende den diskrete cosinustransformation. Men resultaterne var ret dårlige: Fingeraftryk består af små stærkt lokaliserede særkendetegn som små løkker og tvedelinger, der er afgørende for at kunne matche en mulig forbryders fingeraftryk med arkivet. Men netop disse særkendetegn blev hurtigt ret slørede, når man anvendte fourier-analyse. Så også FBI vendte sig mod den moderne wavelet-analyse. Det betyder selvfølgelig ikke at fourier-analysen er på vej ud, men at vi i dag har langt flere analysemetoder og derfor i langt højere grad kan afveje fordele og ulemper ved de enkelte analysemetoder i forbindelse med praktiske anvendelser.