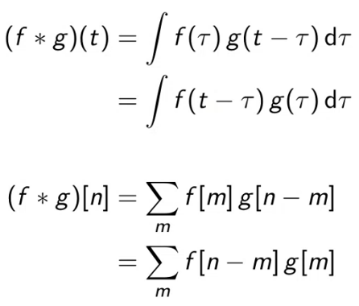
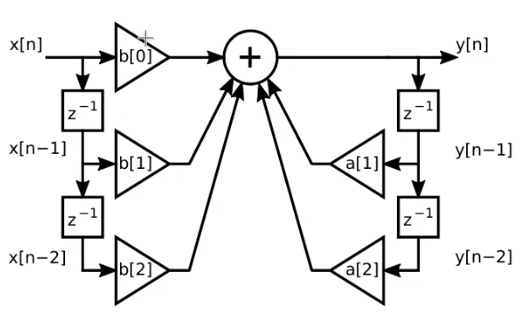
**Integrálne transformácie obrazu**

**Signál**

* Veličina závislá na čase, frekvencie, polohe..
* Obraz je dvojrozmerný signál, kde voľné premenné sú priestorové súradnice *x* a *y*
  + možno transformovať do frekvenčnej domény, ktorá má nejaký horizontálnu a vertikálnu frekvenciu
* Stacionárny vs. nestacionárny signál
  + k analýze stacionárnych signálov stačí furierovú transofrmáciu
  + lenže obraz je nestacionárn, takže potrebujeme robustnejšie filtre
* **Filtrácia**
  + vytiahne z obrázku nejaké komponenty (dolná horná priepusť)
* Lineárne vs. nelineárn filtry

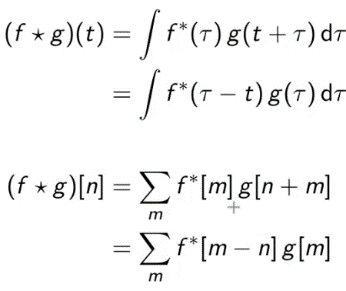
**Lineárna filtrácia**

* Výstup je lineárna kombinácia lineárnych filtrov
* Časovo invariantný filter je taký ktorý vždy na rovnakú sekvenciu odpovie rovnakým výstupom
* Odozva lineárneho časovo invariantného (LIT) filtru sa volá konvolúcia
* Konvolučné jadro sa dá nazvať aj ako filter konečnej odozvy (FIR filter)

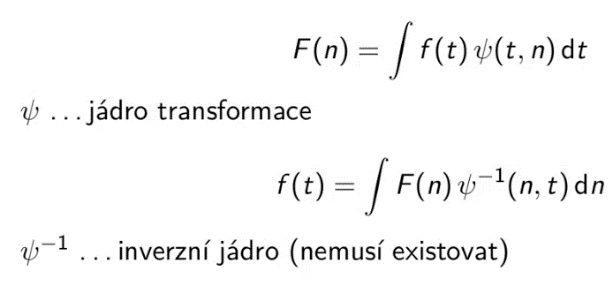
****

**Konvolúcia**

* Konvolúcia zahrňuje pri aplikácia obrátenie jadra, potom nasleduje séria skalárnych súčinov

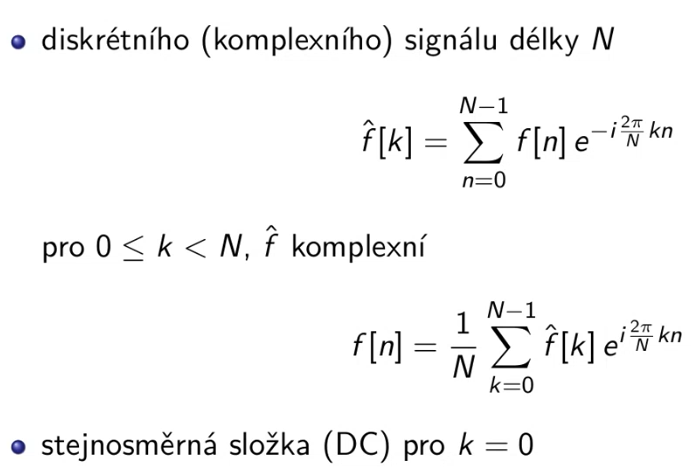
**Korelácia (cross kolerácia)**

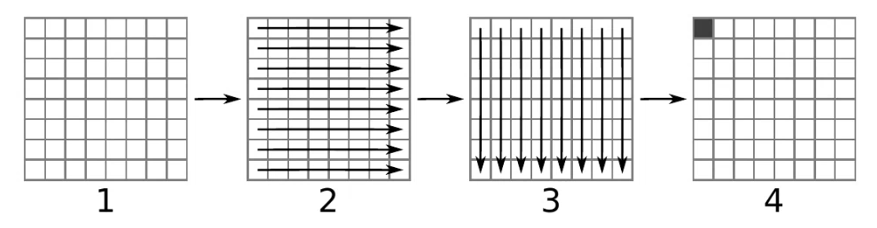
* Podobná operácia ku konvolúcií, rozdielom je len, že pri aplikovaní jadra sa jadro neobráti

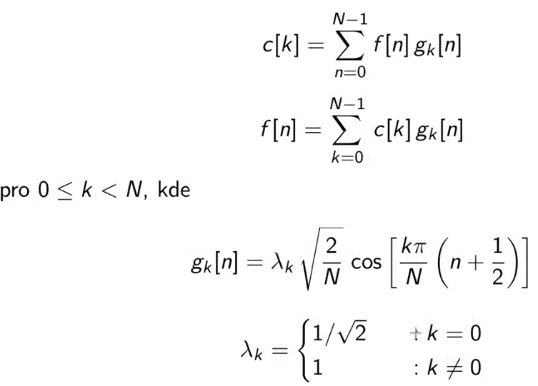
**Integrálna transformácia**

* Vezmem signál *f* a jadro transformácie a urobím itegrál
* V niektorých prípadoch je možne pomocou inverzného jadra získať pôvodný signál

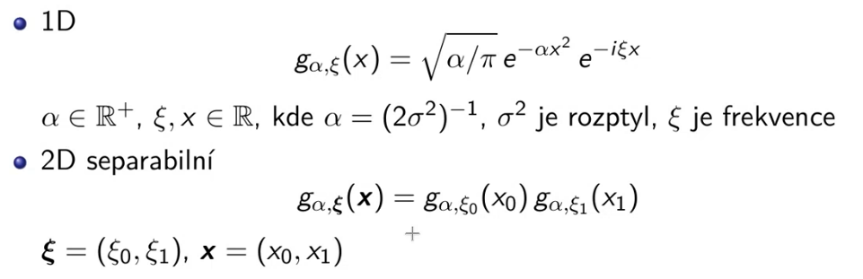
**Diskrétna furierova transformácia (DFT)**

* ****
* Zložtosť O(N­­2), ale existuje algoritmus FFT, ktorý počíta transformáciu linearitmickou zložitosťou O(N log2N)
* Vzhľadom na obrazy nás zaujíma **2D DFT**
  + je separabilná, je možné ju spočítať pomocou série lineárnych operácii



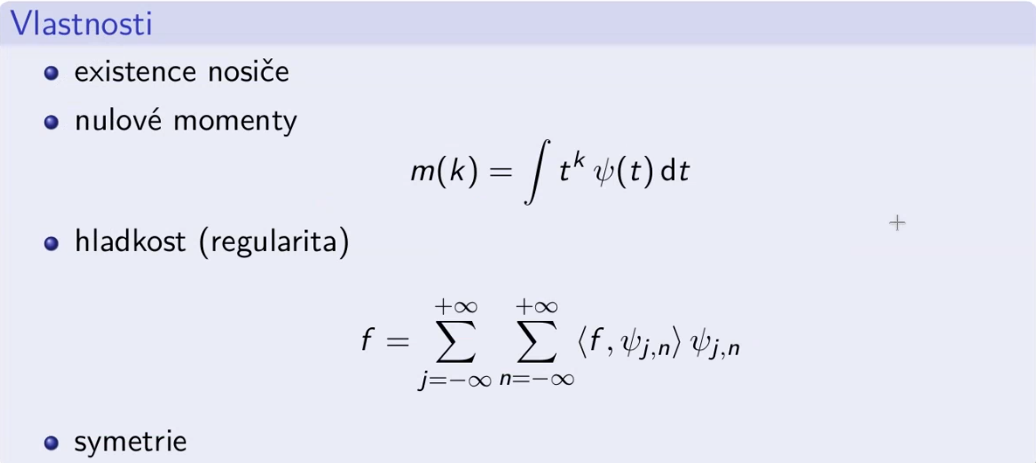
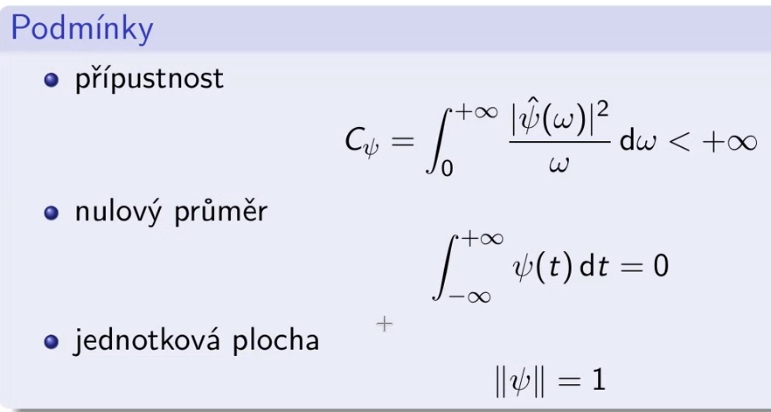


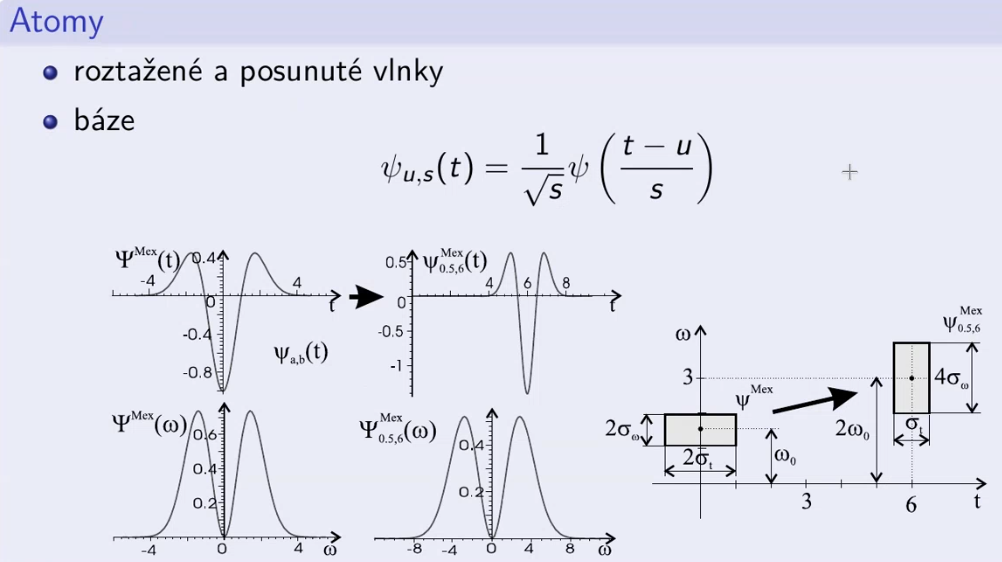
* Využíva sa viaranta DFT – **diskrétna kosínová transformácia (DCT)**
  + tvorí pre reálný signál komplexné koeficienty
  + je symetrická
  + rozloží obraz na základe stavebné kamene *g*, majú tvar kosínusovky
* Pre **2D DCT** plati to isté ako pre 2D DFT
  + tiež je to separabilná operácia, najprv po riadkoch, potom po stĺpcoch..
  + napr. JPEG, aplikuje 2D DCT na bločky 8x8 vzorkov

**2D Gáborova transformácia**

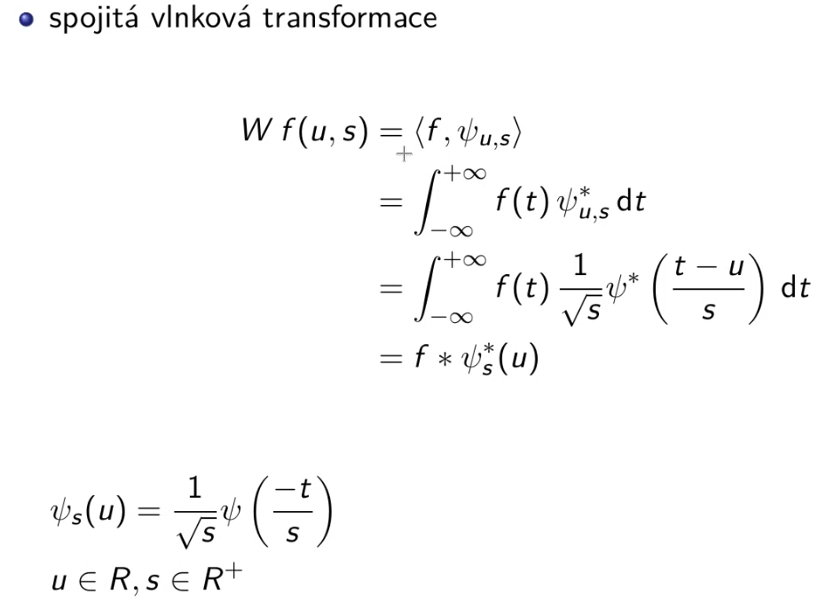
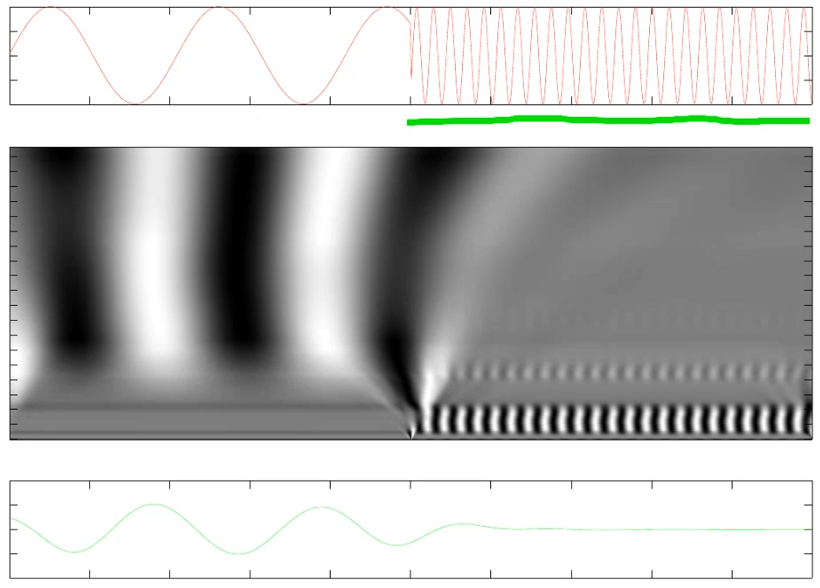
* Extrakcia príznakov, detekcia hrán, segmentácia

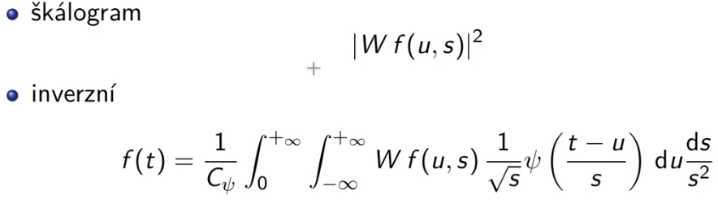
**Vlnka**

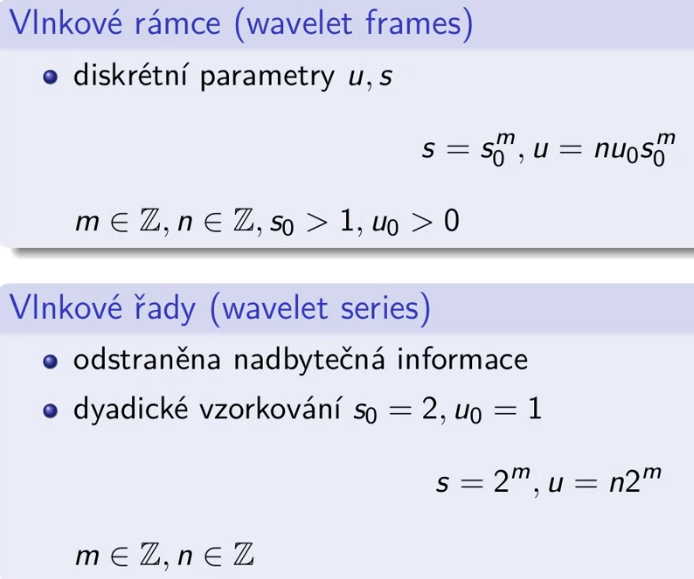
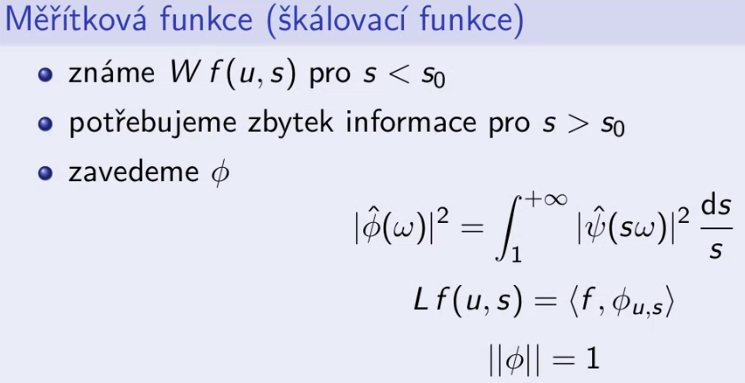




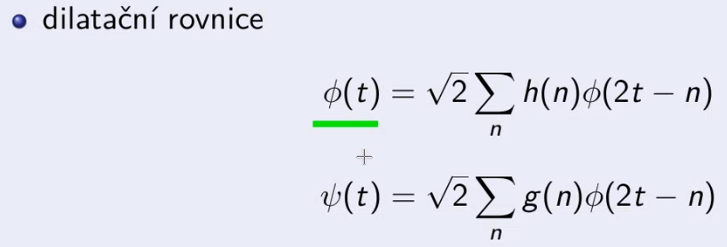
* Jadro transformácie sú rôzne roztiahnuté a posunuté vlnky

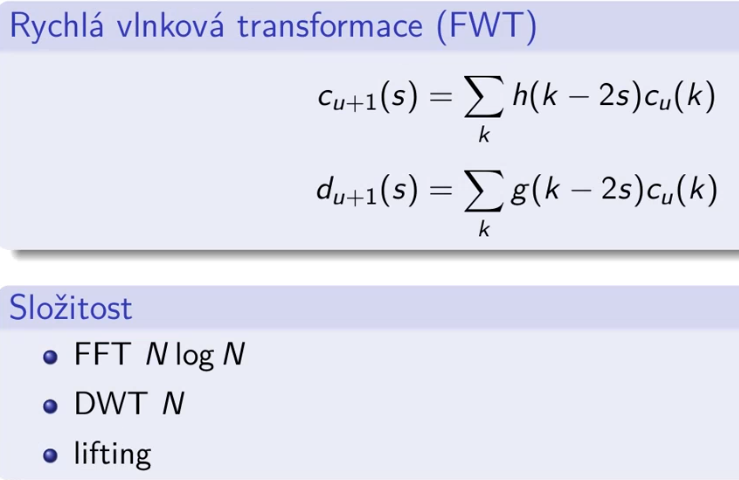


* Vizualizácia

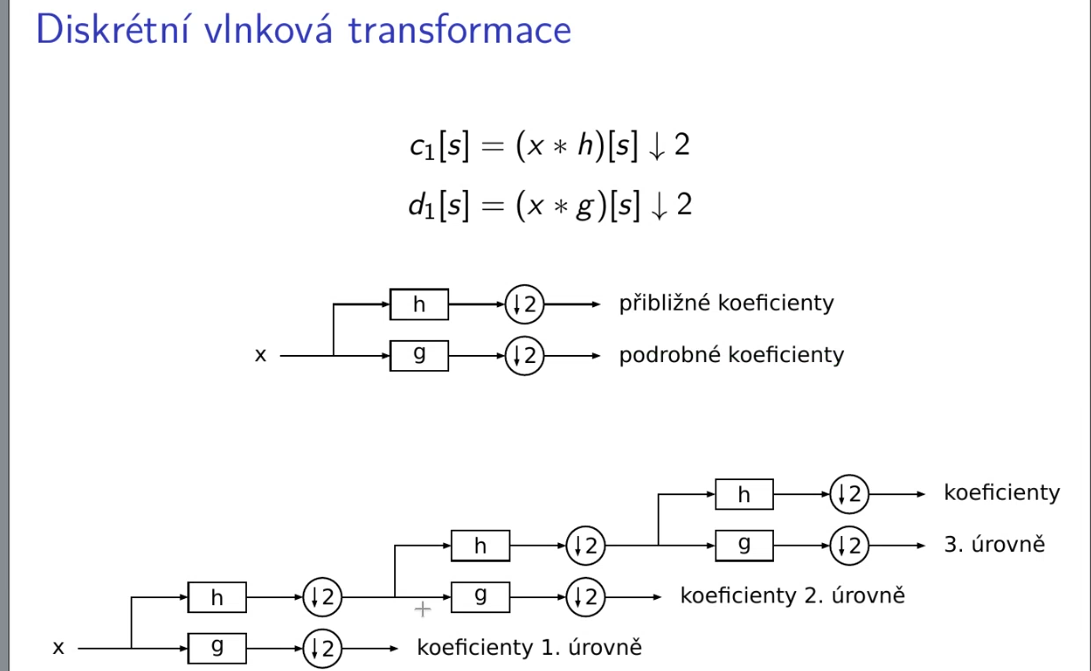


* Zaujíma nás až diskrétna vlnková transofrmácia
* Podstatné je, že mierítková funkcia a vlka je cez diletačné rovnice viazaná s koeficientmi cez dva FIR filtre
* Môžem teda spočítať seriu konvolúcií s nejakými krátkymi FIR filtrami (sú rôzne roztiahnuté a posunuté)



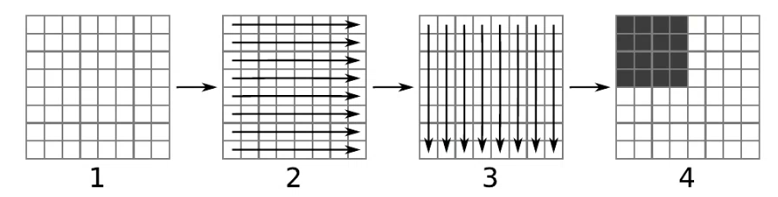


* Lifting je hrozne rýchly systém výpočtu diskrétnej vlnkovej transofrmácie
* Transformácia sa aplikuje hierarchicky typicky

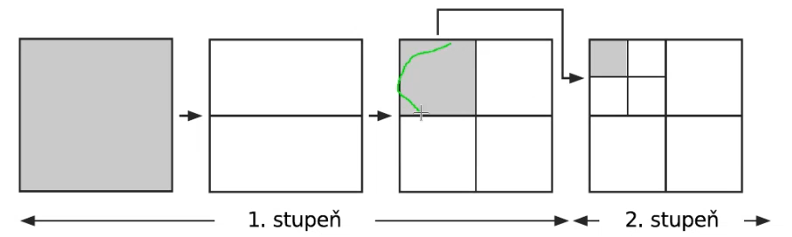


**Nad obrazom**

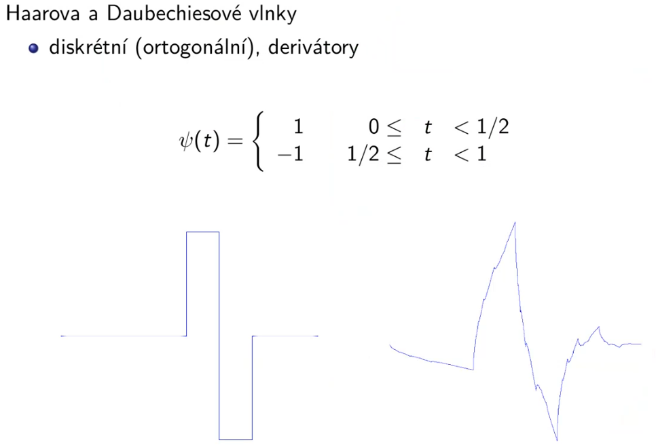
* Rovnako ako do teraz.. riadky.. stĺpce výsledok..

****

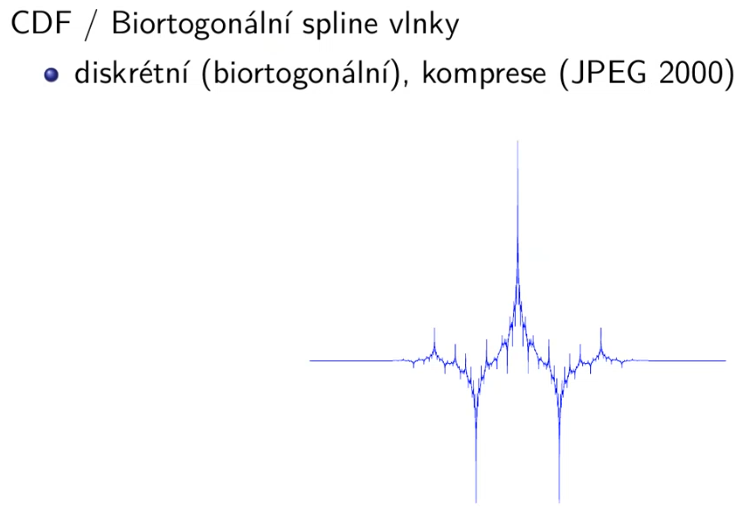
* Po získaní stupňa rozkladu, sa viac zánaram (to čierne je stupeň rozkladu)

****

* Jedna z najdôležitejších transformácií v spracovaní obrazu
* V 1D vzniká vektor
* V 2D vzniká strom reprezentovateľný maticov

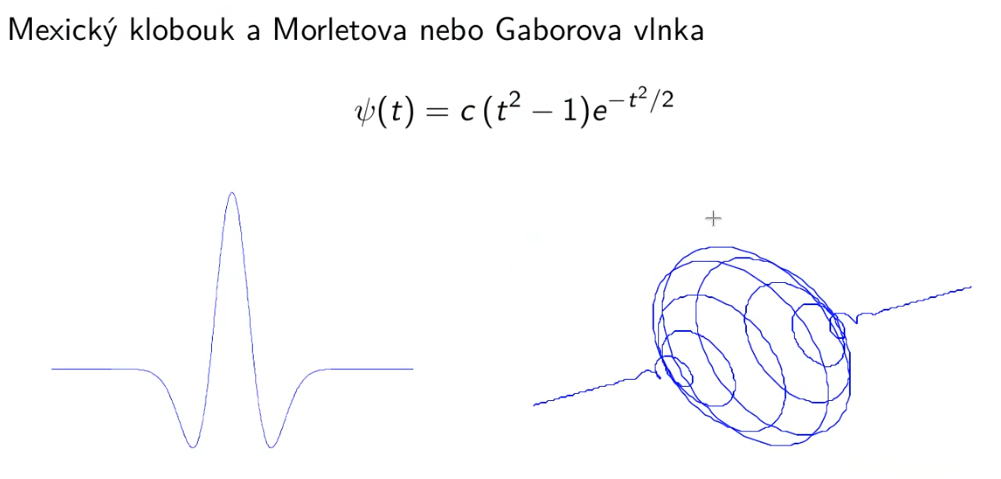
**Používané diskrétne vlnky**

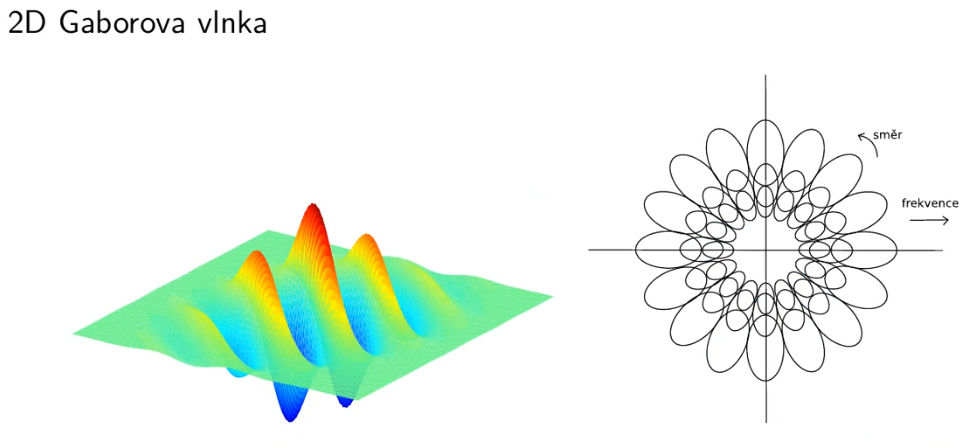
* Haarova je najjednoduhšia, je z rodin daubeschiesových vlniek



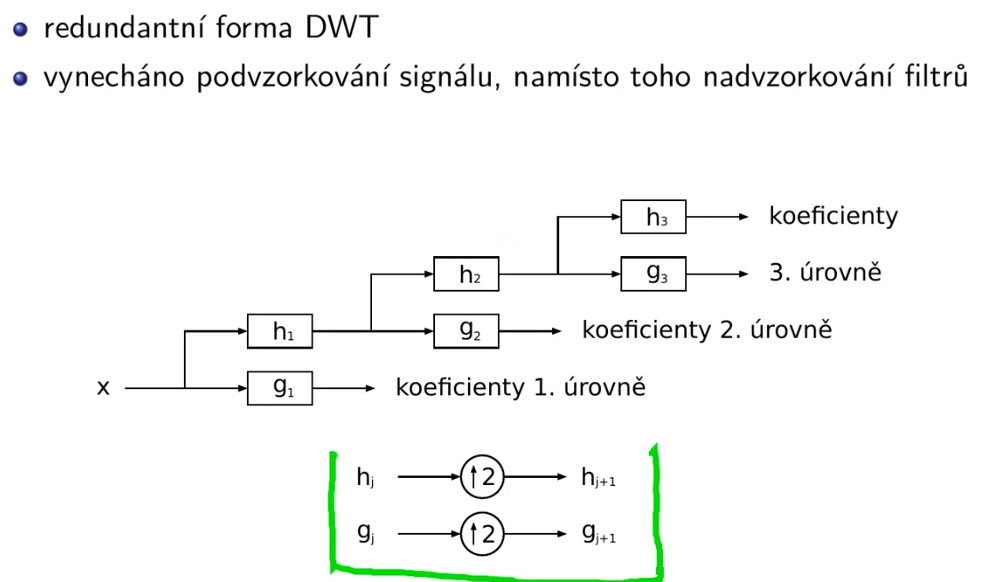
* Použité napr. v kompresii obrazu  JPEG 2000
* Je výpočetne nenáročná, často používaná a symetrická
* Široko sa používa

**Používané spojité vlnky**



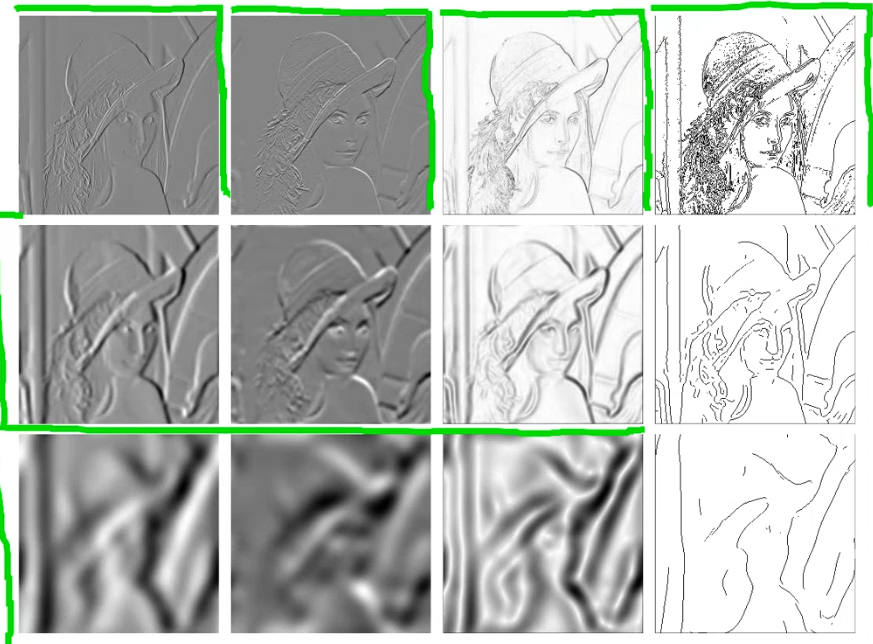


**Stacionárna vlnková transformácia**



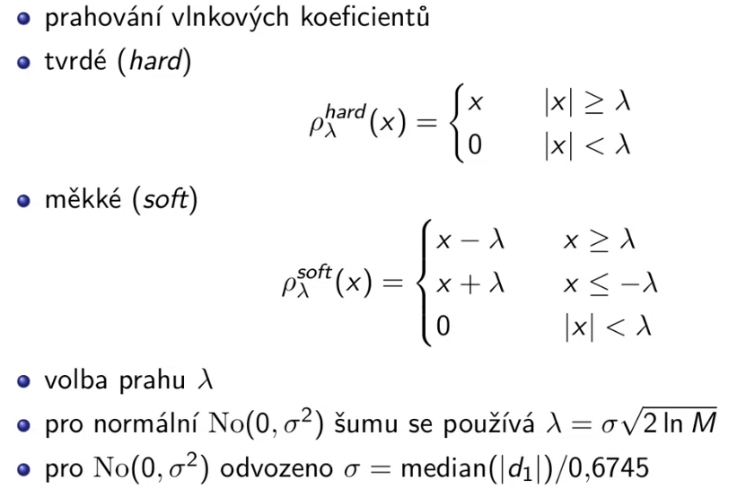
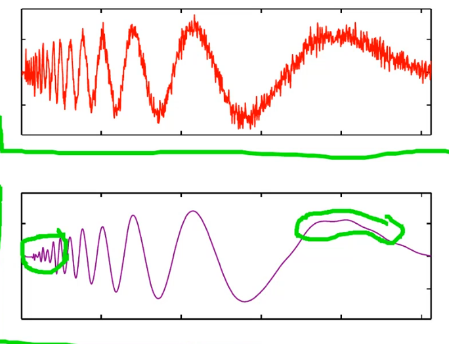
**Detekcia hrán**

* Pomocou teórie vlnkovej transformácie môžem prepnúť akýkoľvek detektor hrán do multiscale detektoru



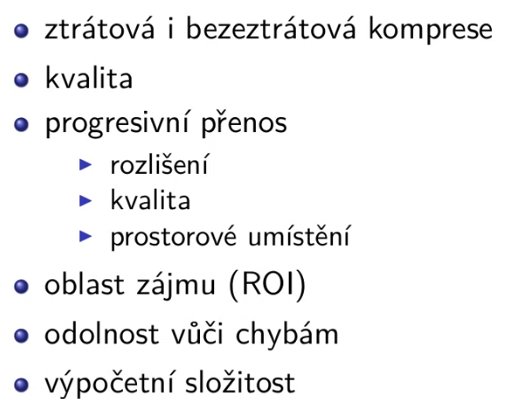
**Odstránenie šumu**

* Bohužiaľ niektorých artefaktov sa nezbavím

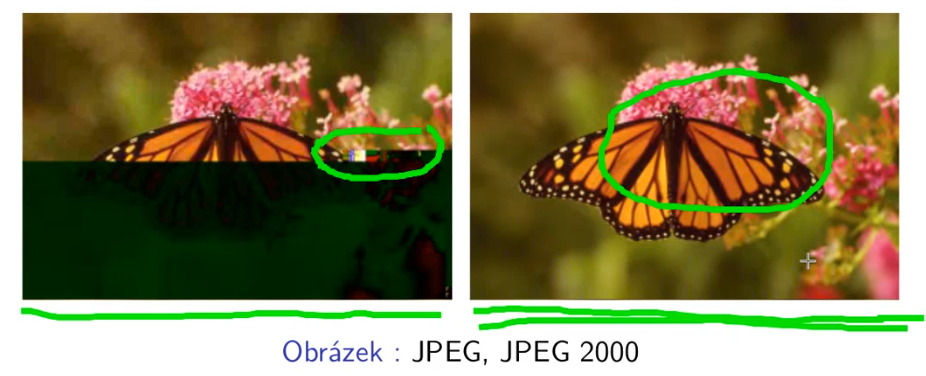
****

**Diskrétna vlnková transformácia**

* Možnosti a výhody:



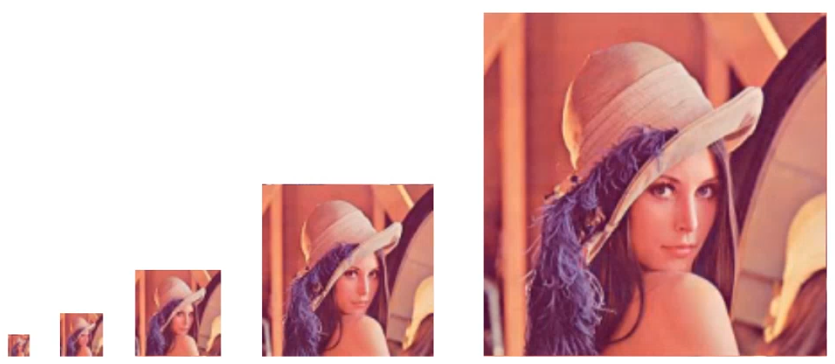
* Toto je už možne pri formáte JPEG 2000 (nie pri JPEG)
* Ukážka: bolo zapísaných 16 bajtov núl do obrazu:



* Ukážka: oblasť záujmu

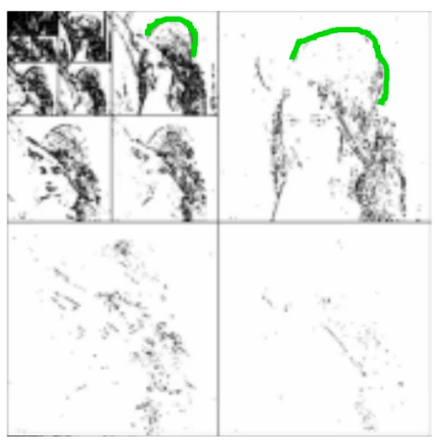


* Ukážka: progresívny prenos (rozlíšenie)

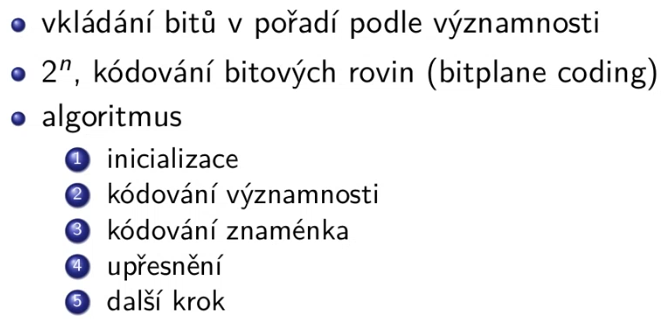


* Ukážka: progresívny prenos (pozícia):



****   
**Korelácia**

**Algorimtus Embedded coder**

****

**Algoritmus EZV**

* Najednoduhší algoritmus

