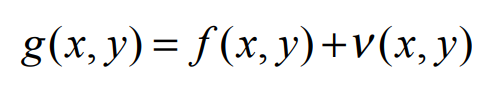
**Chyby v obraze, typy šumu, reštaurácia obrazu a optimálna filtrácia**

**Typy šumu**

* **Z**ákladné delenie:
  + závislý šum
  + nezávislý šum
* Ďaľšie delenie (toto rozdelenie koreluje s predošlím rozdelením)
  + aditívny šum (aj nezávislý šum) – obraz *f* a šum *v* nie sú závislé

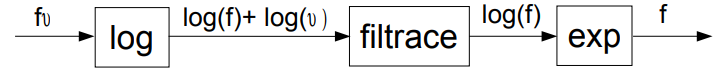


* + multiplikatívny šum (aj závislý šum) – velkosť šumu závisí na veľkosti obrazového signálu



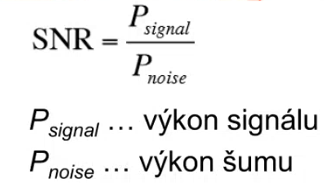
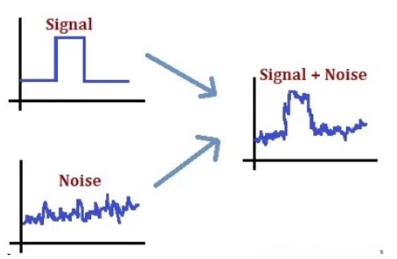
**Homomorfná filtrácia**

* Odstránenie multiplikatívneho šumu z obrázka
* Nie je to dokonalý prístup ale funguje



**Signal-to-Noise Ratio (SNR)**

* Typicky sa uvádza v decibeloch
* Čím vyššia hodnota SNR, tým čistší obrázok
* Meranie pomeru šumu voči obrazovému signálu
* Obrázok bez šumu / šum
* Ak má obrazový signál vyššie amplitúty než šum nie až tak viditeľné poškodenie, ale ak má obraz približne rovnaké amplitúdy ako šum, obraz sa začne javiť značne poškodený

**** ****

* Výkon signálu – suma vzoriek grafu, avšak každá vzorka je umocnená na druhú, aby sa pri sume neodčítali záporné vzorky (potrebujem ako to lieta hore aj dolu..)
  + napr. sčítam všetky hodnoty pixelov na druhú

**Biely šum**

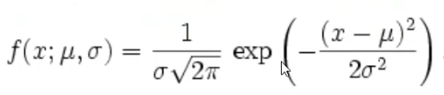
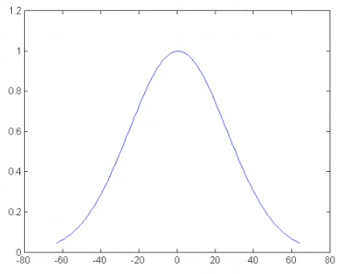
* nekolerovaný šum (náhodný)
* šum má konštantné výkonové spektrum. Intenzita šumu neklesá s rastúcou frekvenciou
* aproximácia najhorších šumov

**„Farby“ šumu**

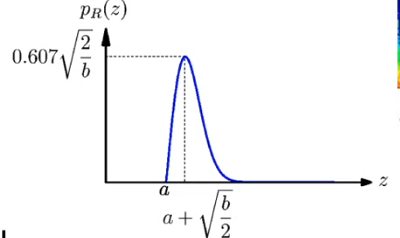


**Gaussov šum**

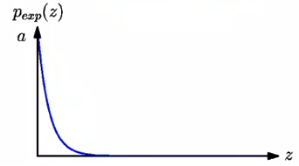
* šum pstihuje všetky pixely obrazu
* amplitúda šumu má normálné (gaussovo) rozloženie pravdepodobnosti



* + stredná hodnota „mí“ je obvykle rovná 0
  + najvyššia pravdepodobnosť je, že pixel nezmení vôbec, vrchol gaussovky je v bode 0
  + málo pravdepodobné je podľa obrázku úprava o 60
* Biely gaussov šum vo veľa situáciách veľmi dobre aproximuje reálny šum
  + napr. šum v prenosovom kanáli
* Síce je vhodný ale často sa používa aj tam, kde by nemal

**Rayleigh šum**

* Tvar rozloženia je trochu iný než gaussov
* Range images – intenzita odpovedá vzdialenosti od kamery, napr. kinekt ho využíva

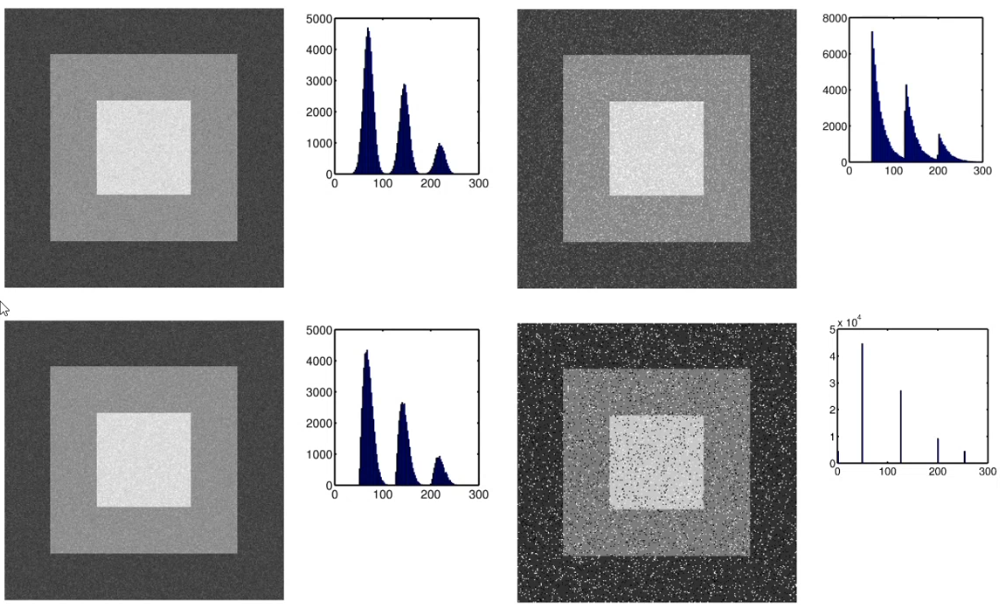
**Exponenciálny šum**

* Laser imaging – lejzrové simanie obrazu

**„Pepř a súl“**

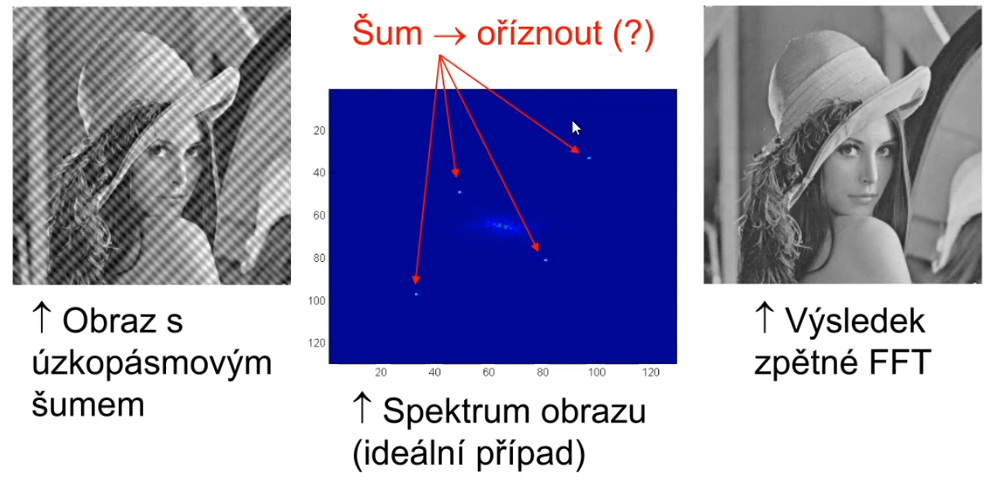
* Často nazývaný aj impulzný šum alebo výstrelový
* Napadne len niektoré pixle obrazu a zmení ich na extrémne hodnoty napr. biela, alebo čierna
* Je nutné na opravu detekovať tieto extrémne body a ich hodnotu upraviť pomocou okolných pixlov
  + napr. vyhľadenie mediánovým filtrom

**Porovnanie šumov**

****

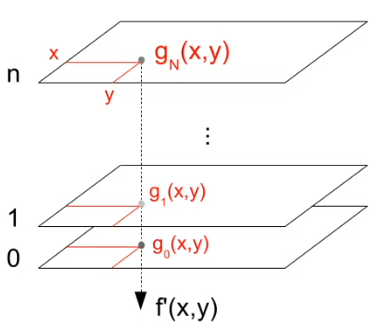
**Úzkopásmový šum**

* postihuje len niektoré frekvencie obrázku (na rozdiel od bieleho, ktorý postihuje všetky frekvencie)
* riešením odstránenia tohto šumu je filtrácia frekvenčnej oblasti

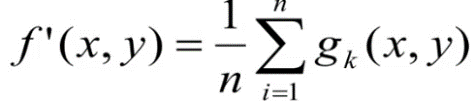


**Štatistický prístup k filtrácií šumu**

* metódy predpokladajú, že je šum aditívny a strednú hodnotu má nula a je nezávislý na signále
  + priemerovanie z viacero snímkov (ale bez rozmazania)
  + priemerovanie z opakujúcich sa oblastí
  + priemerovanie vo vnútri snímku

**Priemerovanie z viacerých snímkov**

* Ak máme n realizácií rovnakej nemennej scény
* Realizácie však nemôžu obsahovať „pohyb“ pixelov, ak to obsahujú je nutné obrazy najprv predspracovať (stabilizácia obrazu)
* Vtedy môžem spriemerovať pixely ktoré sú nad sebou

****

**Priemerovanie jedného snímku**

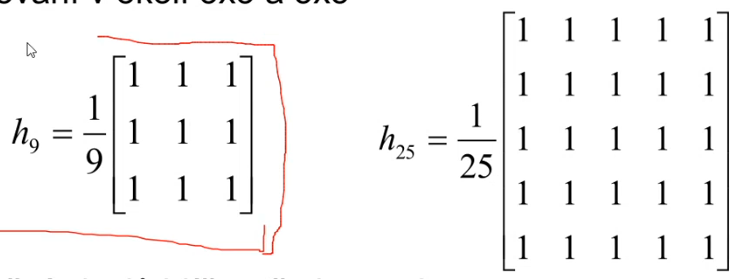
* Často potrebujeme filtrovať šum z jediného obraz -> problem
* **Priemerovanie opakujúcich sa oblastí**
  + V podstate niekoľko realizácií v jednom obraze
* **Priemerovanie vo vnútri snímku**
  + Susedné pixely majú prevažne rovnakú alebo podobnú hodnotu jasu
* Representant okolia alebo priemer z niekoľkých hodnôt
* Problém rozmazávania hrán v obraze

**Lokálne predspracovanie obrazu**

* Rozdelenie:
  + **linéarne filtre** (obyč vyhladzovanie) – možná ľubovoľná kombinácia filtrov či aplikácia v rôznom poradí, výsledok bude vždy rovnaký
  + **nelineárne filtre** – tu záleží na poradí či kombinácií
* **Konvolúcia** – konvolučným oknom prechádzam obrázok, a prenásobím jednotlivé hodnoty konvolučného jadra s obrázok a sčítam ich dokopy
* **Filtrácia vo frekvenčnej oblasti** – prevediem obraz do spektra a tam na ňom pracujem

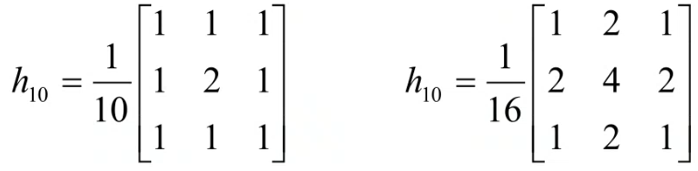
**Priemerovanie**

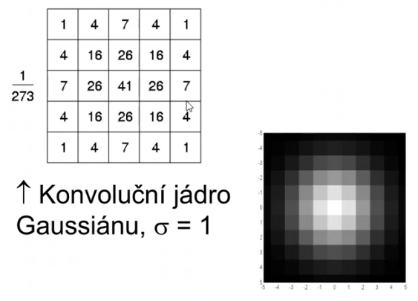
* Ak chceme zachovať intenzitu obrázku, suma filtru musí byť jedna, jednoducho dosiahnuteľné, zvoliť si váhy aké chcem a podeliť ich súčet.
* Príklad priemerovania:



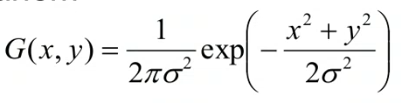
**Zvíraznenie pixlov v strede masky**

* Stále zachovávam sumu váh filtru rovnú jednej, nechcem zvyšovať intenzitu obrazu



** Priemerovanie Gaussian filtrom**

* Lineárny filter, jeho jadro je definované vzťahom

****

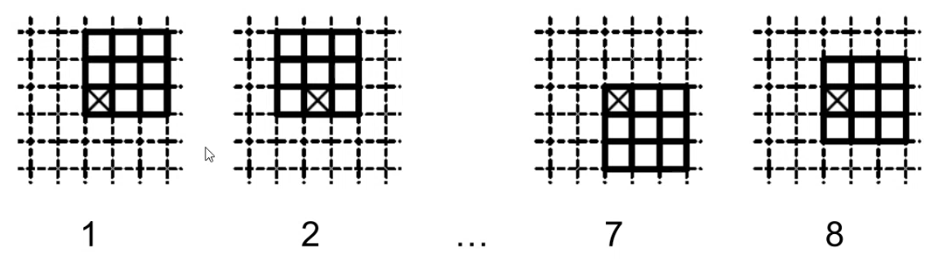
* Chcem aby v strede filtra boli hodnoty najvyššie a v okolí postupne klesali, napr. podľa gaussovho rozloženia

**Nelineárne vyhladzovanie**

* Nechceme vyhladzovať aj hrany ako do teraz
* Stredná hodnota je zlý odhad, pokiaľ existujú vychýlené hodnoty -> medián

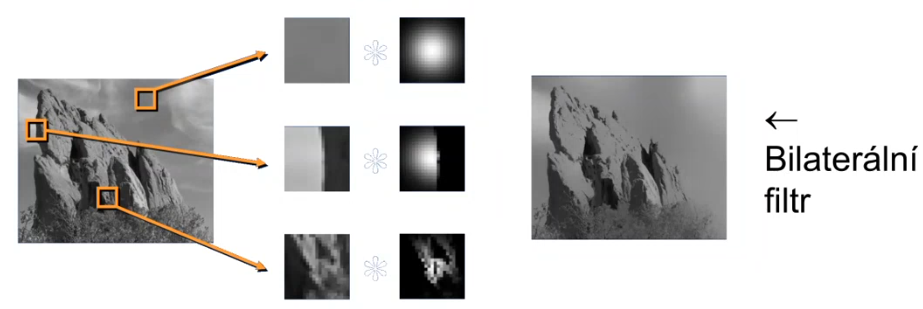
**Filtrácia pomocou rotujúcej masky**

* V okolí 5x5 vyhľadáva homogénnu časť rotujúca maska
* Hľadám pozíciu masky v ktorej je najmenší rozptyl pixelov
* Pre tento prípad mám 9 rôznych polôh

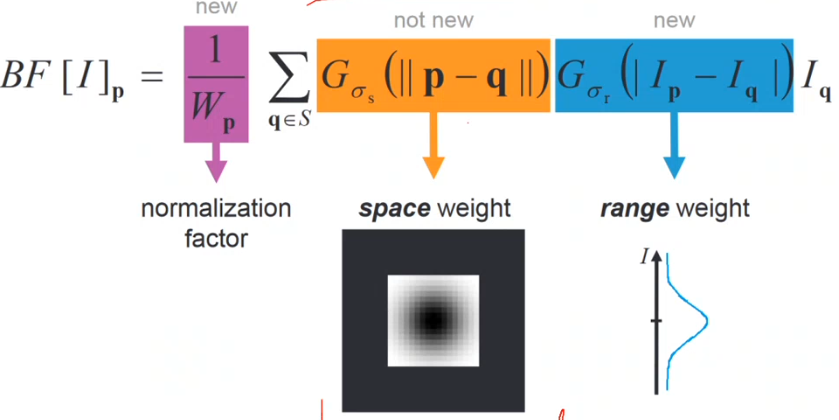
****

**Bilaterálny filter**

* Lepšia varianta rotujúcej masky, rozširuje klasický gaussov filter
* Využíva gaussovo jadro, ale je to nelineárny filter!
* Upravuje gaussovo jadro na základe toho ako pod jadrom obraz vyzerá

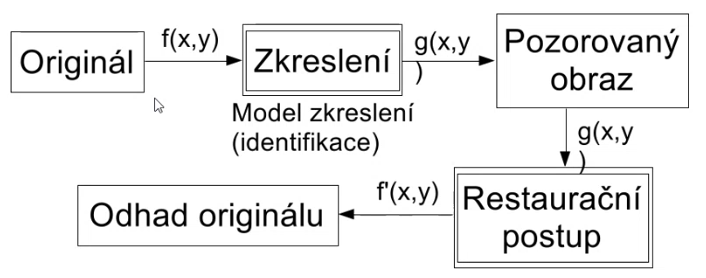


* Váhy okolných pixlov:
  + Maska sa generuje podľa vzdialenosti od stredu okolia (typický gaussov filter)
  + a ešte podľa rozdielu intenzít pixelov od stredu okolia

****

**Reštaurácia (obnovenie) obrazu**

* Rekonštrukcia originálneho obrazu z poškodeného obrazu
* Predpokladá sa známa degradácia obrazu
* Je nutný model zklesnia, pre nás konvolúcia



* Najčastejší lineárny model poruchy obrazu:

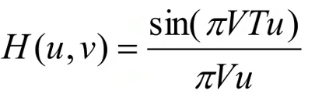


* Prístupy:
  + **deterministické metódy** – predpokladá známu degradáciu, slabý šum, najčastejšie sa použije inverzná transformácia k degradácií obrazu
  + **štatistické metódy** – optimalizačné kritéria, napr suma najmenších štvorcov

**Identifikácia zkreslenia**

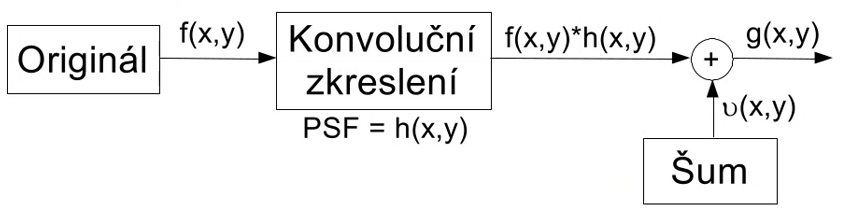
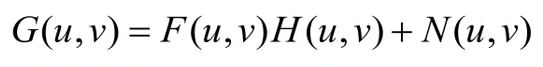
* Analiticky – spočíta sa impulzná charakteristika
* Experimentálne – hviezda (stálica) veľmi ďaleko
* Aposterórna identifikácia – zo skreslených obrázkov, používajú sa líniové a hranové ciele
  + Line Spread Function
  + Step Spred Function

**Rozmazanie pohybom**

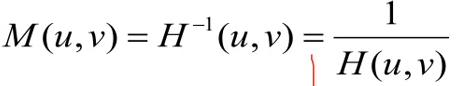
* Objekt sa pohybuje konštantnou rýchlosťou *V* v smere osy *x* vzhľadom ku kamerev dobe otvorenia záverky po dobu *T*
* Analytické riešenie modelu poruchy je:
* Takto odstránim rozmazanie pohybom

**Inverzná filtrácia**

* Inak povedané **dekonvolúcia**
* Lineárny model zkreslenia obrazu

**** ****

* Pri prevedení do spektra sa mi z konvolúcie stane násobenie
* Rekonštručný filter, ak je *N(u,v)* zanedbateľné

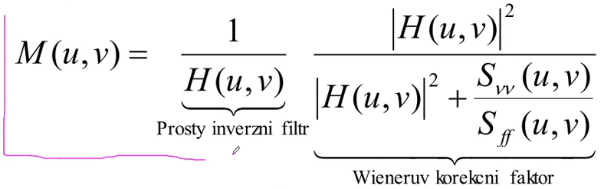
****

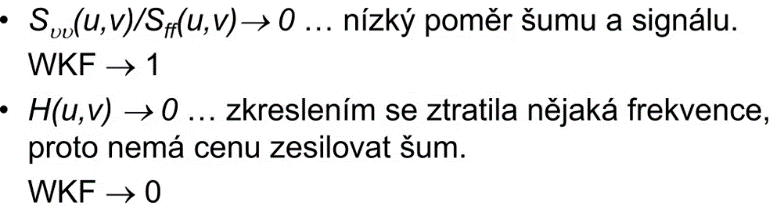
* Lenže ten šum tam je, takže uvedený rekonštručný filter nestačí (ak je zanedbateľný tak nejak to fungovať bude)

**Optimálna filtrácia obrazu**

* Je nutné o šume niečo vedieť, inak to nejde
* Jednotlivé metódy sa líšia tým, aké vlastnosti poznáme
* Metódy:
  + Wienerov filter
  + Viazaná dekonvolúcia – celková energia šumu
  + Slepá dekonvolúcia – nepoznáme nič (dnes je možne natrénovať neuronovkou)

**Wienerov filter**

* Pracuje pre nezanedbateľný šum, ktorý má odhadnuteľné štatistické vlastnosti
* Šum je nezávislý na signále
* Filter:
* Tá veľká čast je wienerov korekčný faktor
  + je v intervale 0 až 1
  + pozerá sa na to, či vo frekvencií *f(u, v)* sa nachádza šum



* Nepoužíva sa tento tvar filtru (len na vysvetlenia sa používa)
* V reálnom vystupuje len výkonové spektrum šumu a obrazu:

