Dolžina vektorja: $||a|| = \sqrt{\sum_{i=1}^{n} a_i^2}$, **Skalarni produkt**: $a \cdot b = \sum_{i=1}^{n} a_i\ b_i$

HIPERSPEKTRALNO ZAJEMANJE

Spektralni kot/kosinusna razdalja (p=spekter piksla, e_i=ciljni spekter): $\alpha = arccos(\frac{p \cdot e_i}{||p||\ ||e_i||})$

popolno spektralno razmesanje: 1. Spektri vseh končnih članov v sceni morajo biti znani. 2. Resi sistem n linearnih enacb za vsak piksel (n-st. koncnih clanov, e_i-i-ti koncni clan, ς-spekter, p-piksel):
$\varsigma(p) = \alpha_1\varsigma(e_1) + \alpha_2\varsigma(e_2) + ... + \alpha_n\varsigma(e_n)$

ODSTRANJEVANJE PEGASTEGA ŠUMA

povprecje: $\mu = \frac{\sum_i^N x_i}{N}$, **standardni odklon**: $\sigma = \sqrt{\frac{\sum_i^N (x_i - \mu)^2}{N}}$, **varianca**: $\sigma^2 = \frac{\sum_i^N (x_i - \mu)^2}{N}$

Adit. šum: $I(p) = X(p) + N(p)$, **Mult. šum**: $I(p) = X(p)N(p)$, I-šumna slika, X-brezšumna slika, N-šum, $p = [i, j]$

MSE $= \frac{1}{MN}\sum_i\sum_j(I(p) - \hat{X}(p))^2$, **RMSE** $= \sqrt{MSE}$, **PSNR** $= 20log_{10}(\frac{(max_p I(p))}{RMSE})[dB]$, **SNR** $= 10log_{10}\frac{\sigma_I^2}{\sigma_X^2}[dB]$

SNR $= \frac{\mu_I}{\sigma_I}[dB]$, **S/MSE** $= 10log_{10}\frac{\sum_i\sum_j \hat{X}(p)^2}{\sum_i\sum_j(I(p)-\hat{X}(p))^2}[dB]$

Ocenjevanje nivoja pegastega šuma v slikah SAR: **ENL** $= (\frac{\mu_I}{\sigma_I})^2 = \frac{1}{\sigma_N} = L$ (šum ima povprecje 1 ($\mu_N = 1$))

Adaptivni mediana filter - 1. Definiramo vecjo maske H_1 in manjšo masko H_2. 2. Vse piksle manjse od praga T filtriramo z masko H_1, preostale pa z masko H_2. 3. Korak 2 lahko izvedemo 2x.

Kuanov filter - predpostavke so: 1. multiplikativni šum: $I(p) = X(p) + (N(P) - 1)X(p)$, 2. šum ima povprecje 1. Postopek: $\hat{X}(p) = \mu_I + \frac{\sigma_X^2(I(p)-\mu_I)}{\sigma_X^2+(\mu_I+\sigma_X^2)/L}$, $\sigma_X^2 = \frac{L\sigma_I^2-\mu_I^2}{L+1}$, izredni primer: $\sigma_X^2 < 0$, potem $\hat{X}(p) = \mu_I$.

Leejev (MMSE) filter: $\hat{X}(p) = \mu_I + \frac{\sigma_X^2(I(p)-\mu_I)}{\sigma_X^2+\mu_I/L}$ **Filter gamma**: $\hat{X}(p) = \frac{(\alpha-L-1)\mu_I+\sqrt{\mu_I(\alpha-L-1)^2+4\alpha L\mu_I I(p)}}{2\alpha}$, $\alpha = \frac{L+1}{L(\sigma_I/\mu_I)^2-1}$

Frostov filter: $h = e^{-KC_I(t_0)||t||}$, $C_I = \frac{\sigma_I}{\mu_I}$ (K - parameter filtra (ponavadi 1), t_0 je lokacija piksla, $||t||$ je razdalja do t_0).

Oddyjev filter (3x3): if $m < \alpha\mu_I$ then $\hat{X}(p) = \mu_I$, else $\hat{X}(p) = \frac{\sum_k\sum_l W_{kl}I(k,l)}{\sum_k\sum_l W_{kl}}$
kjer if $|I(k,l) - I(p)| < m$ then $W_{kl} = 1$, else $W_{kl} = 0$. $m = \frac{1}{8}\sum_k\sum_l|I(k,l) - I(p)|$. α je podan (npr. 1).

Filter AFS - Postopek: 1. Vrednost l.e.s (ploščina) izračunamo za 9 binarnih mask znotraj okna velikosti 5x5. 2. Izberemo masko z minimalno vrednostjo l.e.s. 3. Filtriramo z nizkim sitom izbrano masko. 4. Dobljeno vrednost priredimo središčnemu pikslu okna, velikosti 5x5.

BARVNA VZTRAJNOST

Model za formiranje barvnih vrednosti piksla: $C_v(p) = \int_\lambda S_v(\lambda)E(\lambda, p)d\lambda$ (λ-valovna dolžina, p - piksel, v-barvni kanal)

Poenostavitev zgornje enačbe: $C_v(p) = G(p)R_v(p)L_v(p)$ (R-odbojnost predmetov, L-spekter svetlobe, G-geom. faktor)

Algoritem WPR: $L = max(C(p))$, nato: $O(p) = G(p)R(p) = \frac{C(p)}{L}$

Algoritem GWA: $O(p) = \frac{C(p)}{\overline{C}\varphi}$ ($\overline{C}$ - povp. vrednost barvnega kanala, φ - faktor povečave(npr. 2))

Algoritem CCN - iteriramo: 1. $O'(p) = \frac{C(p)}{C_R(p)+C_G(p)+C_B(p)}$, 2. $O(p) = \frac{O'(p)}{\overline{C}\varphi}$ 3. $C(p) = O(p)$

Algoritem LSAC: $O(p) = \frac{C(p)}{L(p)} = \frac{C(p)}{\overline{C}(p)\varphi}$, $\overline{C}(p) = C(p) * K(p)$, $K(p) = K(x, y) = a^{-1}e^{-\frac{x^2+y^2}{2p^2}}$, $p = 0.093max(M, N)$ (K-jedro, MxN slika)

Algoritem RSR: P mnozic, vsaka vsebuje K pikslov. $p_{p,k} = [i_{p,k}, j_{p,k}]$-k-ti piksel p-te mnozice.
$i_{p,k} = i + r\eta cos(\mu)$, $j_{p,k} = i + r\eta sin(\mu)$, $n \in U(0, 1)$, $\mu \in U(0, 2\pi)$, $p = [i, j]$, r-radij razpršene množice.
$O(p) = \frac{1}{P}\sum_{p=1}^{P}\frac{C(p)}{max_k(C(p_{p,k}))}$

POSPLOŠENA HOUGHOVA TRANSFORMACIJA

Splošna analitična krivulja: $F(x, a) = 0$, kjer x - piksel, a - vektor parametrov krivulje

<table>
<tr><th>Postopek HT za iskanje linij oz. premic</th><th>Postopek HT za iskanje krožnic, na osnovi robnih pikslov</th></tr>
<tr><td>
Klasicna enačba premice: $y = kx + n$
Parametrična oblika enačbe: $xcos(\theta) + ysin(\theta) = r$
Pri θ izračunamo oddaljenost r za piksel $p = [i, j]$:
$r = icos(\theta) + jsin(\theta)$ (enacba 2)
1. Inicializiraj akumulator A, tj. $A(\theta, r) = 0$
2. FOR vsak robni piksel:
    Izračunaj enačbo (2)
    $A(\theta, r) = A(\theta, r) + \Delta A$
3. Poišči lokalne maksimume v akumulatorju A, kar ustreza položajem premic v sliki.
</td><td>
Klasična enačba krožnice: $(x - a)^2 + (y - b)^2 = r^2$
1. Inicializiraj akumulator A, tj. $A(a, b, r) = 0$
2. FOR vsak robni piksel
    FOR $a = min_a : \Delta a : max_x$
        FOR $b = min_b : \Delta b : max_b$
            $r = \sqrt{(x - a)^2 + (y - b)^2}$
            $A(a, b, r) = A(a, b, r) + \Delta A$
<hr>
HT elipse, na osnovi robnih pikslov - enačba elipse:
$\frac{(x-x_0)^2}{a^2} + \frac{(y-y_0)^2}{b^2} = 1$ (4 params, 3 FOR-loops),
z dodano rotacijo po x osi pa:
$x(t) = x_0 + a\ cos(t)\ cos(\theta) - b\ sint\ sin(\theta)$ in
$y(t) = y_0 + a\ cos(t)\ cos(\theta) + b\ sint\ sin(\theta)$ (5 params, 4 loops)
</td></tr>
</table>

HT za analitične krivulje v sivinskih slikah, na osnovi vseh informacij iz slike robov (jakost + smer gradienta)	**HT za iskanje elips v sivinskih slikah, s pomočjo informacije o smeri gradienta**

Originalna enačba krivulje $F(x,a)=0$ (enacba 3)
Njen odvod: $\frac{dF}{dx}(x,a)=0$ (enacba 4)
V odvodu dobimo znan clen $\frac{dy}{dx}=tg[\psi(x)-\frac{pi}{2}]$
Postopek:
1. Inicializiraj akumulator A, tj. A(a) = 0
2. FOR vsak robni piksel
    Izračunaj parametre a, za katere velja (3) in (4)
    $A(a)=A(a)+\Delta A$
3. Poisci lokalne maksimume v akumulatorju A.
	Enačba elipse: $\frac{(x-x_0)^2}{a^2}+\frac{(y-y_0)^2}{b^2}=1$ (enacba 5)
	$X=x-x_0$ in $Y=y-y_0$, odvajamo po X
	in dobimo $\frac{2X}{a^2}+\frac{2Y}{b^2}\frac{dY}{dX}$
	Vrednost $\frac{dY}{dX}$ poznamo, oznacimo $\frac{dY}{dX}=\xi$
	Izračunamo: $Y=\pm\frac{b^2}{\sqrt{1+\frac{a^2}{b^2}\xi^2}}$ in $X=\pm\frac{a^2}{\sqrt{1+\frac{b^2}{a^2\xi^2}}}$
	Dolocimo: $y_0=y\pm\frac{b^2}{\sqrt{1+\frac{a^2}{b^2}\xi^2}}$ in $x_0=x\pm\frac{a^2}{\sqrt{1+\frac{b^2}{a^2\xi^2}}}$
	$\xi=tg(\phi-\varphi-\frac{\pi}{2})$ (enacba 9)
	Postopek:
	1. Inicializiraj akumulator A,
	tj. $A(x_0,y_0,\varphi,a,b)=0$
	2. FOR vsak robni piksel
	  FOR $a=min_a:\Delta a:max_a$
	    FOR $b=min_b:\Delta b:max_b$
	      FOR $\varphi=min_\varphi:\Delta\varphi:max_\varphi$
	        Izračunaj enacbo 9
	        Doloci X in Y po (7), pri cemer doloci predznak na osnovi opazovanja dY in dX.
	        Rotiraj X in Y za kot φ
	        $x_0=x+X$ in $y_0=y+Y$
	        $A(x_0,y_0,\varphi,a,b)=A(x_0,y_0,\varphi,a,b)+\Delta A$
	3. Poisci lokalne maksimume v akumulatorju A, kar ustreza polozajem elips v sliki.

Posplositev HT za neanaliticne krivulje - vektor parametrov $a=[y,s,\varphi]$, kjer $y=[x_0,y_0]$ - ref. izhodisce za krivuljo, φ - orientacija krivulje, $s=[s_x,s_y]$ - ortog. skalirna faktorja (v praksi enaka), R-tabela=tabela moznih orientacij robnih pikslov (za opis ref. izhodisca y).

Postopek kreiranja R-tabele	**Oblika R-tabele**

1. Izberi referencno točko y znotraj iskane oblike.
2. FOR vsako točko x iz meje
    Izračunaj smer gradienta $\psi(x)$
    $\vec{r}=\vec{y}-\vec{x}$
    Shrani r v R-tabelo v odvisnosti od $\psi(x)$.
Postopek iskanja poljubnih oblik:
1. Inicializiraj akumulator A, tj. $A(y,s,\varphi)=0$
2. FOR vsak robni piksel x
  FOR $s=min_s:\Delta s:max_s$
    FOR $\varphi=min_\varphi:\Delta\varphi:max_\varphi$
      Doloci R' po enacbi (10).
      Vstavi R' v enacbo (11) in doloci R''.
      FOR $r\in R''(\psi(x))$
        $A(x+r,s,\varphi)=A(x+r,s,\varphi)+\Delta A$
3. Poisci lokalne maksimume v akumulatorju A
	i | ψ_i | R_{ψ_i}
	0 | 0 | $\{r \mid y-r=x\ \wedge\ \psi(x)=0\ \wedge\ x$ je na meji $\}$
	1 | $\Delta\psi$ | $\{r \mid y-r=x\ \wedge\ \psi(x)=\Delta\psi\ \wedge\ x$ je na meji $\}$
	2 | $2\Delta\psi$ | $\{r \mid y-r=x\ \wedge\ \psi(x)=2\Delta\psi\ \wedge\ x$ je na meji $\}$
	... | ... | ...
	Lastnosti:
	1. Skaliranje oblike za faktor s: $R'(\psi)=sR(\psi)$ (10)
	2. Rot. za kot φ: $R'(\psi)=Rotacija\{R[(\psi-\varphi)\ mod\ 2\pi],\varphi\}$ (11)
	3. Sprememba ref. točke iz y v y': $R'(\psi)=R(\psi)+(y-y')$
	
	Alternativne inkrementalne strategije: $A(a)=A(a)+\lvert grad(x)\rvert$ ali $A(a)=A(a)+\lvert grad(x)\rvert+konstanta$

MODELI AKTIVNIH KONTUR

Položaj kače: $v(s)=[x(s),y(s)]$, kjer je s proporcionalen dolzini loka, tj. $s\in[0,1]$. **Energijski funkcional kace**: $E_{snake}=\int_0^1(E_{int}(v(s))+E_{image}(v(s))+E_{con}(v(s)))ds$, kjer E_{int} - **interna energija zlepka zaradi upogibanja**, E_{image} - **slikovne sile**, E_{con} - **sile zunanjih omejitev**. $E_{int}=\frac{1}{2}(\alpha(s)|v_s(s)|^2+\beta(s)|v_{ss}(s)|^2)$, kjer v_s - prvi odvod v po s (clen 1. reda), v_{ss} - drugi odvod v po s (clen 2. reda), α in β - utezi (ponavadi $\alpha=\beta$). $E_{image}=w_{line}E_{line}+w_{edge}E_{edge}+w_{term}E_{term}$ (13), kjer $E_{line}=I(x,y)$ - energ. funkcional, ki pritegne kačo k linijam. $E_{edge}=-|\Delta I(x,y)|^2$ - energ. funk., ki pritegne kačo k robovom. Zglajena slika: $C(x,y)=G_\sigma * I(x,y)$. Torej je **energ. funk. zaključkov**: $E_{term}=\frac{C_{yy}C_x^2-2C_{xy}C_xC_y+C_{xx}C_y^2}{(C_x^2+C_y^2)^{\frac{3}{2}}}$. C_x,C_y,C_{xx},C_{yy} in $C_{x,y}$ so parcialni odvodi. **Diskretizacija energije kače**: $E_{snake}=\sum_{i=0}^n E_{int}(i)+E_{ext}(i)$. i-to vozlisce kace: $v_i=[x_i,y_i]=[x(ih),y(ih)]$. E_{int} postane: $E_{int}(i)=\alpha_i\frac{|v_i-v_{i-1}|^2}{2h^2}+\beta_i\frac{|v_{i-1}-2v_i+v_{i+1}|^2}{2h^4}$. Naj bo $F_x(i)=\frac{\partial E_{ext}}{\partial x_i}$ in $F_y(i)=\frac{\partial E_{ext}}{\partial y_i}$. Dobimo Eulerjevo enacbo: $\alpha_i(v_i-v_{i-1})-\alpha_{i+1}(v_{i+1}-v_i)+\beta_{i-1}(v_{i-2}-2v_{i-1}+v_i)-2\beta_i(v_{i-1}-2v_i+v_{i+1})+\beta_{i+1}(v_i-2v_{i+1}+v_{i+2})+(F_x(i),F_y(i))=0$. To zapisemo v matricni notaciji: $Ax+F_x(x,y)=0$ in $Ay+F_y(x,y)=0$ (x in y sta n-dim. vektorja x in y koord. kače). To resimo kot $x_t=(A+\gamma I)^{-1}(\gamma x_{t-1}-F_x(x_{t-1},y_{t-1}))$ in $y_t=(A+\gamma I)^{-1}(\gamma x_{t-1}-F_y(x_{t-1},y_{t-1}))$ (enacbi 16). γ je velikost koraka (pogosto 1).

Pentadiagonalna matrika A		**Algoritem kac**

$\begin{bmatrix} cba0000...0ab \\ bcba000...00a \\ abcba00...000 \\ 0abcba0...000 \\ .... \\ 000000..abcba \\ 000000..0abcb \\ 000000..00abc \end{bmatrix}$	$a=\beta$
	$b=-(\alpha+4/\beta)$
	$c=2\alpha+6\beta$
	
	zgornje velja, ce $\alpha_i=\alpha$ in $\beta_i=\beta$
		1. Vnesi zacetni priblizek (krivuljo) iskanega objekta.
		2. Enakomerno razporedi n vozlišč po krivulji.
		3. Določi E_{ext} po enacbi (13).
		4. Izračunaj $F_x=\frac{\partial E_{ext}}{\partial x}$ in $F_y=\frac{\partial E_{ext}}{\partial x}$
		5. Doloci matriko A
		6. WHILE (ni izpolnjen konvergencni pogoj)
		    Doloci nov polozaj vozlisc kace v iteraciji t po enacbi (16).
		    $t=t+1$
		    Enakomerno razporedi n vozlisc po krivulji.

Iskanje ujemanja v stero slikah - dodatni energijski funkcional: $E_{stereo}=(v^L(s)-v^R(s))^2$

ANALIZA NEODVISNIH KOMPONENT $x_j(t) = a_{j1}s_1(t) + a_{j2}s_2(t) + \cdots + a_{jn}s_n(t) \quad \forall j$

Mešalni model: $\underbrace{\mathsf{X}}_{n\times N} = \underbrace{\mathsf{A}}_{n\times n}\underbrace{\mathsf{S}}_{n\times N}$ => $\mathsf{S} = \mathsf{WX}$, kjer je
n - število opazovanih spremenljivk in neodvisnih komponent
N – število vzorcev
A – mešalna matrika
W – inverz matrike A

Mere za merjenje "ne Gaussov"
Kurtozis: $kurt(y) = E\{y^4\} - 3(E\{y^2\})^2 \overset{\text{pred.}}{=} E\{y^4\} - 3$, kjer je
y - naključna spremenljivka z ničelnim povprečjem ter varianco 1
E – matematično upanje
Negentropija: $J(y) = [E\{\mathcal{G}(y)\} - E\{\mathcal{G}(v)\}]^2$, kjer je
v – Gaussova spremenljivka z ničelnim povprečjem ter varianco 1
G – poljubna nekvadratična funkcija
$\mathcal{G}(u) = \frac{1}{a}\log\cosh au \quad \mathcal{G}(u) = -e^{-\frac{u^2}{2}}$ a – konstanta iz intervala $[1, 2]$

Beljenje
$E\{\mathbf{x}\mathbf{x}^{\mathbf{T}}\} = \mathsf{EDE}^T$, kjer je
E – ortogonalna matrika lastnih vektorjev
D – diagonalna matrika pripadajočih lastnih vrednosti
Beljenje: $\tilde{\mathsf{X}} = \mathsf{D}^{-\frac{1}{2}}\mathsf{E}^T\mathsf{X}$ Razbeljenje: $\mathsf{A} = \mathsf{ED}^{\frac{1}{2}}\tilde{\mathsf{A}}$

Algoritem ICA

Postopek za en stolpec inverzne mešalne matrike W

1. Naključno določi stolpec $\mathbf{w}$ (dimenzije $n \times 1$).
2. Izračunaj
$$\mathbf{w}^+ = E\{\mathsf{X}\mathcal{G}'(\mathsf{X}^T\mathbf{w})\} - E\{\mathcal{G}''(\mathsf{X}^T\mathbf{w})\}\mathbf{w}$$
3. $\mathbf{w} = \mathbf{w}^+/\|\mathbf{w}^+\|$
4. IF $\left((\mathbf{w}^T\mathbf{w} - 1) < \epsilon\right)$ Konec; ELSE Korak 2

Postopek za celotno matriko

1. FOR vsak stolpec matrike $\widetilde{\mathsf{W}}$
   Izvedi postopek za en stolpec (točka 1).
2. Dekoreliraj vrstice v matriki $\widetilde{\mathsf{W}}$, npr. kot
   $\widetilde{\mathsf{W}} = \widetilde{\mathsf{W}}/\sqrt{\|\widetilde{\mathsf{W}}\widetilde{\mathsf{W}}^T\|}$
   REPEAT
   $\widetilde{\mathsf{W}} = \frac{3}{2}\widetilde{\mathsf{W}} - \frac{1}{2}\widetilde{\mathsf{W}}\widetilde{\mathsf{W}}^T\widetilde{\mathsf{W}}$
   UNTIL rezultat ne konvergira
3. Določi originalno mešalno matriko A z razbeljenjem:
$$\mathsf{A} = \mathsf{ED}^{\frac{1}{2}}\widetilde{\mathsf{W}}^{-1}$$