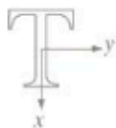







TUTTE LE FORMULE INTERAZIONE E MULTIMEDIA

Formula per rappresentazione di immagini usando la luce incidente (i) e quella riflessa (r)

$$f(x, y) = i(x, y)r(x, y)$$

Formule delle trasformazioni affini

Transformation Name	Affine Matrix, T	Coordinate Equations	Example
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = w$	
Scaling	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = c_x v$ $y = c_y w$	
Rotation	$\begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v \cos \theta - w \sin \theta$ $y = v \sin \theta + w \cos \theta$	
Translation	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$	$x = v + t_x$ $y = w + t_y$	
Shear (vertical)	$\begin{bmatrix} 1 & 0 & 0 \\ s_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v + s_v w$ $y = w$	
Shear (horizontal)	$\begin{bmatrix} 1 & s_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = s_h v + w$	

Forward mapping (T è la matrice affine)

$$[xy1] = [vw1] * T$$

Inverse mapping (T è la matrice affine)

$$[vw1] = [xy1] * inversa(T)$$

Formula Interpolazione Bilineare

$$v(x, y) = ax + by + cxy + d$$

Formula Interpolazione Bicubica

$$v(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} x^i y^j$$

Formula del MSE

$$MSE = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N [I'(x, y) - I(x, y)]^2$$

Formula del PSNR

$$PSNR = 10 \log_{10} \left(\frac{S^2}{MSE} \right)$$

Nell'occhio i coni sono: 6/7 milioni

I bastoncelli 75/150 milioni

Raggio foro "reale" del pinhole

$$r = \sqrt{\lambda d}$$

Equazione lente sottile

$$\frac{1}{u} + \frac{1}{v} = \frac{1}{f}$$

Formula magnificazione

$$m = \frac{v}{u}$$
$$f = \frac{um}{(m + 1)}$$

Nyquist Rate è il doppio della frequenza massima nel segnale

Formula quantizzazione uniforme

$$L' = \frac{L * K}{N}$$

Formula quantizzazione NON uniforme

$$L' = \frac{f(L) * K}{f(N)}$$

oppure generalizzando:

$$L' = f(L, N, K)$$

Una quantizzazione non uniforme è quella logaritmica

$$L' = \frac{\log_2 L * K}{\log_2 N}$$

Formule per i valori del tristimolo

$$X = \int_{380}^{780} L_{e,\lambda} \bar{x}(\lambda) d\lambda$$

$$Y = \int_{380}^{780} L_{e,\lambda} \bar{y}(\lambda) d\lambda$$

$$Z = \int_{380}^{780} L_{e,\lambda} \bar{z}(\lambda) d\lambda$$

Formula per trovare la z nel diagramma cromatico CIE (2D)

$$z = 1 - (x + y)$$

Formule per passare dai valori del tristimolo XYZ a xyz

$$x = X/(X + Y + Z)$$

$$y = Y/(X + Y + Z)$$

$$z = Z/(X + Y + Z)$$

Formule per passare da CIE XYZ allo spazio colore CIELAB

$$L^* = 116 \left(\frac{Y}{Y_n} \right)^{1/3} - 16$$

$$a^* = 500 \left[\left(\frac{X}{X_n} \right)^{1/3} - \left(\frac{Y}{Y_n} \right)^{1/3} \right]$$

$$b^* = 200 \left[\left(\frac{Y}{Y_n} \right)^{1/3} - \left(\frac{Z}{Z_n} \right)^{1/3} \right]$$

X_n , Y_n e Z_n definiscono il punto bianco

Formula per la distanza Euclidea (tra 2 colori in CIELAB)

$$\Delta E_{ab}^* = \sqrt{(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2}$$

Formule per chroma e hue angle in L^*C^*h avendo il colore in CIELAB

$$C^* = \sqrt{(a^*)^2 + (b^*)^2}$$

$$h = \tan^{-1} \left(\frac{b^*}{a^*} \right)$$

I colori sicuri per il web si esprimono esclusivamente con questi numeri esadecimali
33, 66, 99, CC, FF

Formule per passare da RGB a YUV

$$Y = 0.299R + 0.587G + 0.114B$$

$$U = -0.169R - 0.331G + 0.5B$$

$$V = +0.5R - 0.419G - 0.081B$$

Formule per passare da YUV a YC_bC_r

$$Y = 0.299R + 0.587G + 0.114B$$

$$C_b = U + 128$$

$$C_r = V + 128$$

Per gli istogrammi

Per un immagine $I[m, n]$ si ha $H(k)$ = numero di pixel di valore k

Formula normalizzazione dell'istogramma (problemi di range)

$$v_{\text{nuovo}} = 255 \cdot \frac{(v_{\text{vecchio}} - \min_{\text{osservato}})}{(\max_{\text{osservato}} - \min_{\text{osservato}})}$$

Algoritmo di equalizzazione

$$p_r(r_k) = \frac{n_k}{MN}, \quad k = 0, 1, 2, \dots, L-1$$

$$s_k = T(r_k) = (L-1) \sum_{j=0}^k p_r(r_j) = \frac{(L-1)}{MN} \sum_{j=0}^k n_j, \quad k = 0, 1, 2, \dots, L-1$$

- Sia data una immagine a 3 bit ($L=8$) con 64×64 pixel ($MN=4096$) con la seguente distribuzione di intensità:

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

- Applicando la formula si ha:

$$s_0 = T(r_0) = 7 \sum_{j=0}^0 p_r(r_j) = 7p_r(r_0) = 1.33$$

$$s_1 = T(r_1) = 7 \sum_{j=0}^1 p_r(r_j) = 7p_r(r_0) + 7p_r(r_1) = 3.08$$

$$s_2 = 4.55, s_3 = 5.67, s_4 = 6.23, s_5 = 6.65, s_6 = 6.86, s_7 = 7.00.$$

Arrotondando:

$$s_0 = 1.33 \rightarrow 1 \quad s_4 = 6.23 \rightarrow 6$$

$$s_1 = 3.08 \rightarrow 3 \quad s_5 = 6.65 \rightarrow 7$$

$$s_2 = 4.55 \rightarrow 5 \quad s_6 = 6.86 \rightarrow 7$$

$$s_3 = 5.67 \rightarrow 6 \quad s_7 = 7.00 \rightarrow 7$$

Operazione generica su un immagine

$$g(x, y) = T[f(x, y)]$$

Operatori puntuali:

Negativo

$$255 - f(x, y)$$

Trasformazione logaritmica

$$g(x, y) = c \cdot \log(1 + f(x, y))$$

Trasformazione di potenza

$$g(x, y) = c \cdot (f(x, y))^\gamma$$

La binarizzazione sceglie un valore soglia: i pixel al di sotto diventano neri quelli al di sopra bianchi

FINE ARGOMENTI PROVA IN ITINERE 1