FACOLTA' DI INGEGNERIA

INGEGNERIA INFORMATICA

A.A. 2008/2009

Corso

VISIONE E PERCEZIONE

Docente

Prof. FIORA PIRRI

Tutor

MATIA PIZZOLI

MAPPA DI DISPARITA'

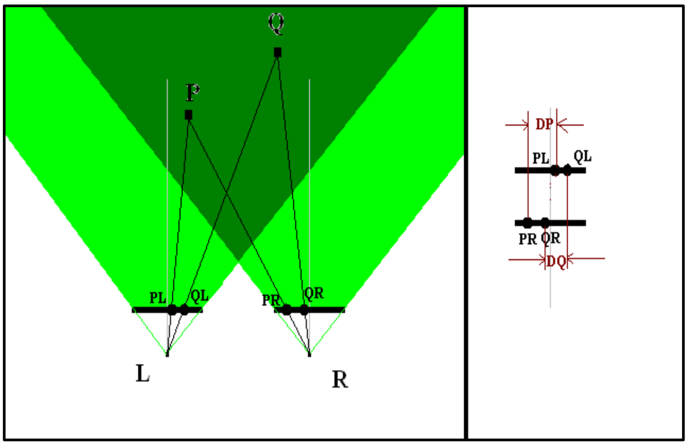
Studente

Redjan Shabani

(1013173)

Definizione di Mappa di Disparità (MD)

Consideriamo due immagini ottenute tramite un sistema stereoscopico[1], chiamate rispettivamente L (immagine sinistra) ed R (immagine destra). Prendiamo in esame lo spazio dei punti del piano contenente gli assi ottici delle telecamere. Supponiamo che sono d'interesse nella scena i due punti P e Q, giacenti nel piano, come indicato anche in figura.



La retta identificata dai punti P ed L interseca il piano immagine sinistro nel punto PL. Cosi anche la retta identificata dai punti Q ed L, interseca il piano immagine sinistro nel punto QL. Analogamente definiamo i punti PR e QR nel piano immagine destro.

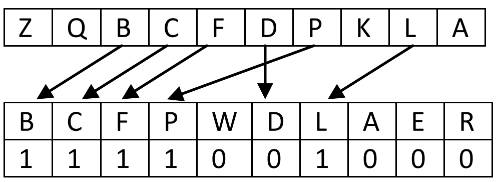
Con elementari considerazioni geometriche si può dimostrare che il punto P (Q) ha come corrispondente destro il punto PR (QR) ottenuto come PL (QL) a sinistra, spostato di DP (DQ) a sinistra. La quantità DP (DQ) e funzione della distanza che ha il punto P (Q) dal piano immagine. La grandezza DP (DQ) è chiamata *disparità* del punto P (Q). La *mappa di disparita* è una matrice delle disparità di tutti i punti dell'immagine sinistra, data l'immagine destra[2]. La mappa di disparità potrebbe essere usata come strumento per la valutazione tridimensionale dell'ambiente osservato.

Principi generali sull'implementazione

L'algoritmo pretende di ricevere in input una coppia di immagini a scala di grigi provenienti da un qualche dispositivo stereoscopico. Si presume inoltre che le telecamere della telecamera stereoscopica siano allineate orizzontalmente.

In linea teorica, data una finestra dell'immagine sinistra, si deve ricercare in tutta l'immagine destra la corrispondenza. Tenendo pero conto delle assunzioni fate all'inizio, si riduce il campo di ricerca nella sola riga dell'immagine destra corrispondente alla finestra presa dall'immagine sinistra[3].

Un altro punto forte nella riduzione dello spazio di ricerca è la proprietà che le corrispondenze a destra dell'elemento di sinistra sono spostati di una certa quantità(la disparità) a sinistra. Inoltre se una regione del segmento di destra è stato già assegnato come corrispondenza di qualcosa a sinistra, non si deve riconsiderare.



In figura viene mostrata un istanza del problema. Sono state effettuate 6 ricerche di corrispondenza, e ogni zona assegnata è stata marcata. Quando si dovrà ricercare la corrispondenza di A ci si deve limitare a confrontare la casella che contiene A nel primo vettore con le caselle contenenti rispettivamente, R, E, A nel secondo vettore.

Linee guida sull'implementazione

Esistono due macroclassi di algoritmi per il calcolo della mappa di disparità:

1. Algoritmi a mappa densa (DMA).
2. Algoritmi a mappa sparsa (SMA).

Il più semplice DMA è quello di effettuare la ricerca delle corrispondenze su ogni pixel dell'immagine sinistra, data l'immagine destra. Ovviamente e molto facile avere delle corrispondenze errate se si considera un singolo pixel come regione da ricercare, per questo il la regione da ricercare non deve essere mai composta da un singolo pixel ma di norma da almeno tre, il pixel centrale e i due vicini laterali. Si può impostare un parametro da passare al algoritmo, che definisce la lunghezza della regione(segmento). Gli SMA, di norma, non eseguono la ricerca su tutti i pixel dell'immagine. Si estraggono una serie di punti, come i corner di Harris, e effettua il calcolo della disparità, come nel caso della mappa densa, per questi punti. In pratica si cerca di implementare algoritmi misti.

Implementazione orientata al gradiente

L'algoritmo riceve in input le due immagini stereo (L,R) e per ciascuna calcola il gradiente.

$$[L_x; L_y] = \left[\frac{\partial L(x,y)}{\partial x} ; \frac{\partial L(x,y)}{\partial y}\right]$$

$$[R_x; R_y] = \left[\frac{\partial R(x,y)}{\partial x} ; \frac{\partial R(x,y)}{\partial y}\right]$$

Calcolati i gradienti si elaborano i moduli:

$$G_L = \sqrt{L_x^2 + L_y^2}$$

$$G_R = \sqrt{R_x^2 + R_y^2}$$

Il calcolo della mappa di disparità viene eseguito sui moduli dei gradienti. Un approccio gradient-oriented e spiegato anche in [1], comunque più complesso e con differenti finalità. In questa sede si è scelto di usare il gradiente partendo semplicemente dall'idea che in un pixel dell'immagine gradiente sono incluse informazioni sull'intorno. Di seguito per L ed R si intendono i gradienti delle immagine sinistra e destra.

In [2] si descrive una tecnica, differente da quella classica, per il calcolo della mappa. Si prende l'immagine destra shiftandola a destra di una colonna, per poi eseguire la differenza assoluta fra immagine sinistra L e immagine destra R shiftata.

R, immagine destra

1	2	3	3	4	5	5
8	7	6	7	4	9	4
9	9	8	7	6	6	6
6	6	8	4	2	6	1
4	7	9	7	4	3	2

S(R,3), immagine destra shiftata di 3 colonne

0	0	0	1	2	3	3
0	0	0	8	7	6	7
0	0	0	9	9	8	7
0	0	0	6	6	8	4
0	0	0	4	7	9	7

Questa differenza viene eseguita per shifting incrementali fino ad un limite, di norma 60. Si ottiene alla fine una collezione di matrici

$$D = \{D_1, D_2, \dots, D_{d_{max}}\}$$

Con $D_i = |L - S(R, i)|$, cioè la differenza fra matrice sinistra e matrice destra shiftata a destra di *i* colonne. Il generico elemento della matrice di disparità MD e dato come segue:

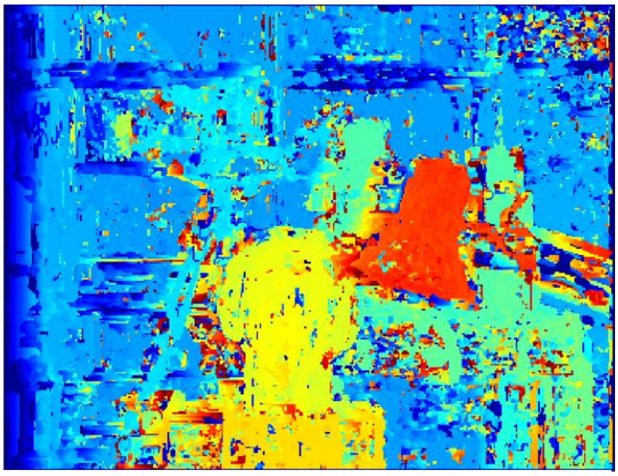
$$MD(i,j) = \underset{k}{\operatorname{argmin}}\{D_k\} = \underset{k}{\operatorname{argmin}}\{|L - S(R,k)|\}$$

cioè l'indice k che rende minima la differenza assoluta fra L ed S(R,k). L'indice k cosi trovato è appunto la disparità del pixel (i,j).

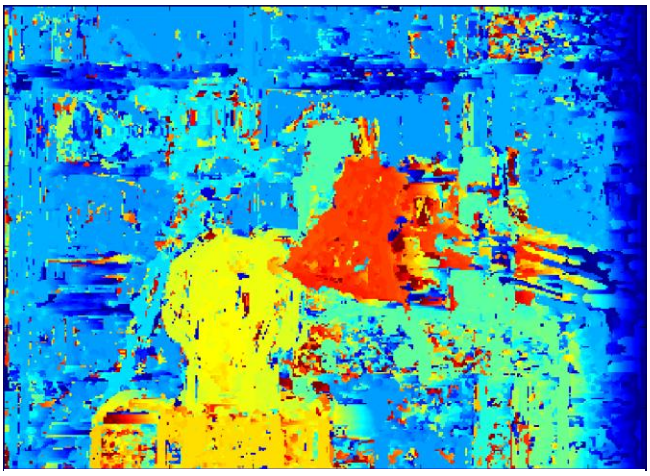
Le trattazioni fatte riguardano immagini a scala di grigi. Per aumentare la precisione del calcolo si è scelto di trattare solo immagini a colori. Il modulo del gradiente di una delle immagini ha tre componenti, cosi come tre sono i componenti del colore. La differenza fra i punti è calcolata come segue:

$$\boldsymbol{D_k(i,j)} = \left(\boldsymbol{L_{\rho(i,j)}} - S(\boldsymbol{R_\rho}, k)_{(i,j)}\right)^2 + \left(\boldsymbol{L_{\gamma(i,j)}} - S(\boldsymbol{R_\gamma}, k)_{(i,j)}\right)^2 + \left(\boldsymbol{L_{\beta(i,j)}} - S(\boldsymbol{R_\beta}, k)_{(i,j)}\right)^2$$

Giustamente all'interno dell'algoritmo non ci si limita a calcolare la differenza solo sui corrispondenti punti (i,j) ma sulla somma degli intorni dei punti (i,j). L'intorno è scelto di grandezza 3-per-3.



Mappa di disparità, riferita all'immagine sinistra, calcolata con l'algoritmo orientato al gradiente.



Mappa di disparità, riferita all'immagine destra, calcolata con l'algortimo rientato al gradiente.

Implementazione orientata al tensore di struttura

Fra le diverse tecniche proposte in [2], per il calcolo della mappa di disparità, c'è anche un algoritmo basato sul tensore di struttura.

Data una matrice(immagine grigia)

$$I = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{bmatrix}$$

e il suo gradiente:

$$[I_x; I_y] = \left[\frac{\partial I(x,y)}{\partial x}; \frac{\partial I(x,y)}{\partial y}\right]$$

si definisce *tensore di struttura non lineare esteso* del generico elemento $a_{i,j}$ di A, la seguente matrice:

$$T = \begin{bmatrix} T_{xx} & T_{xy} & T_{xz} \\ T_{yx} & T_{yy} & T_{yz} \\ T_{zx} & T_{zy} & T_{zz} \end{bmatrix} = \begin{bmatrix} I_x \\ I_y \\ I \end{bmatrix} \cdot [I_x \quad I_y \quad I] = U_s^T \cdot U_s$$

dove I, I_x, I_y si assumono filtrate con una gaussiana.

Per le immagini a colori, posto

$$I' = \frac{1}{3}(I_\rho + I_\gamma + I_\beta)$$

U_s e definita cosi:

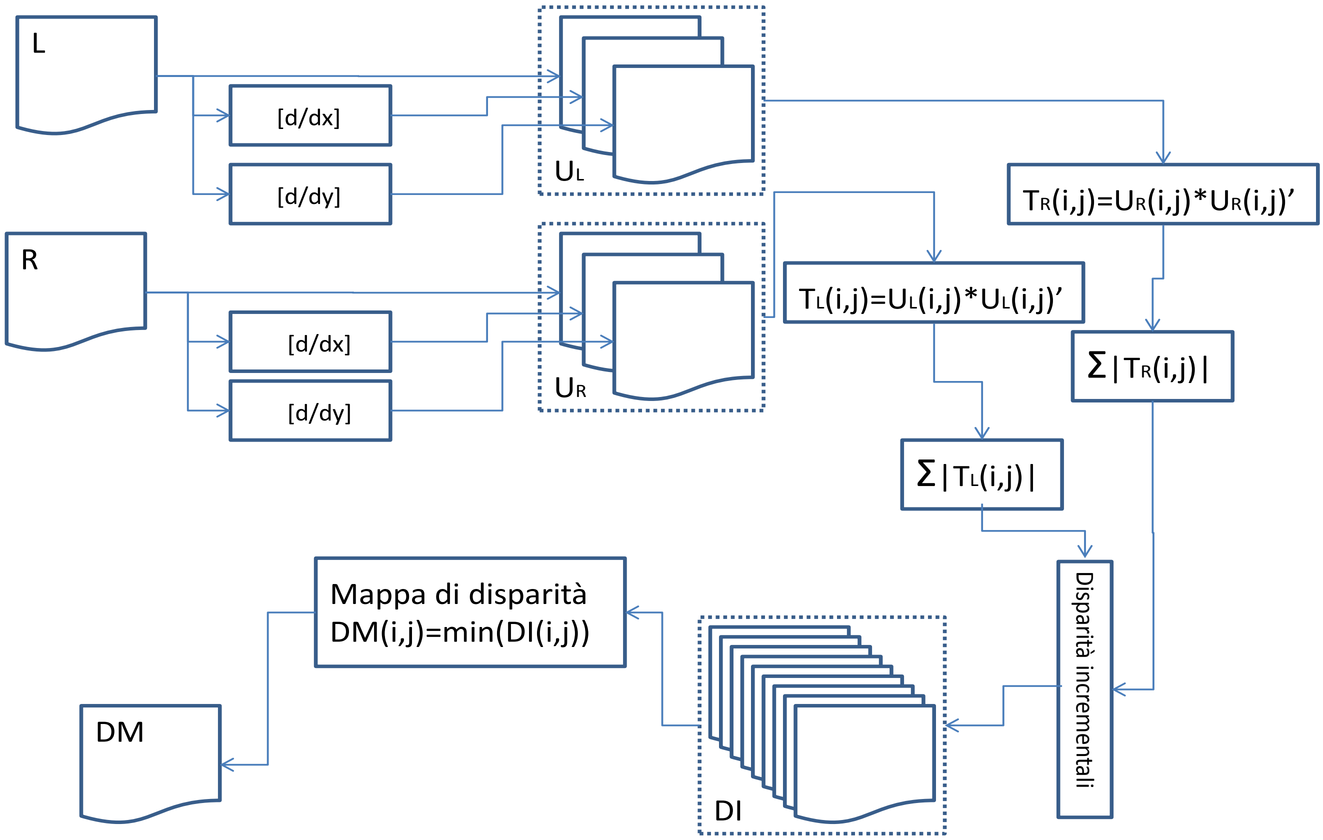
$$U_s = \begin{bmatrix} I'_x \\ I'_y \\ I_\rho \\ I_\gamma \\ I_\beta \end{bmatrix}$$

L'idea alla base di questo approccio deriva dal fatto che il tensore contiene più informazioni del singolo pixel. Dai valori del tensore si possono dedurre informazioni sugli edge o i corner.

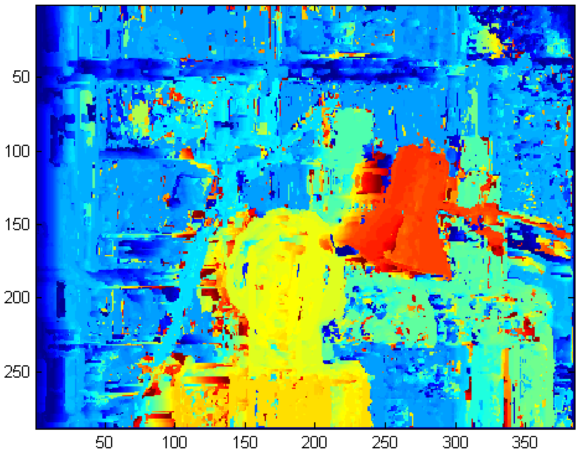
E' stato scelto come misura di differenza tra due punti(tensori di punti) la somma delle differenze assolute fra gli elementi dei tensori:

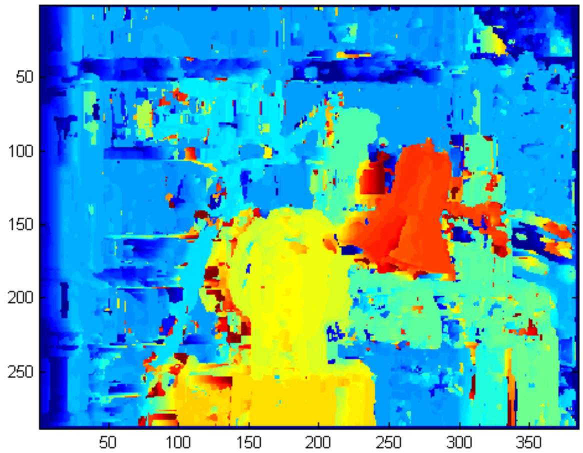
$$dif(T,T') = dif\left(\begin{bmatrix} t_1 & t_2 & t_3 \\ t_4 & t_5 & t_6 \\ t_7 & t_8 & t_9 \end{bmatrix}, \begin{bmatrix} t'_1 & t'_2 & t'_3 \\ t'_4 & t'_5 & t'_6 \\ t'_7 & t'_8 & t'_9 \end{bmatrix}\right) = \sum_{i=1}^{9} |t_i - t'_i|$$

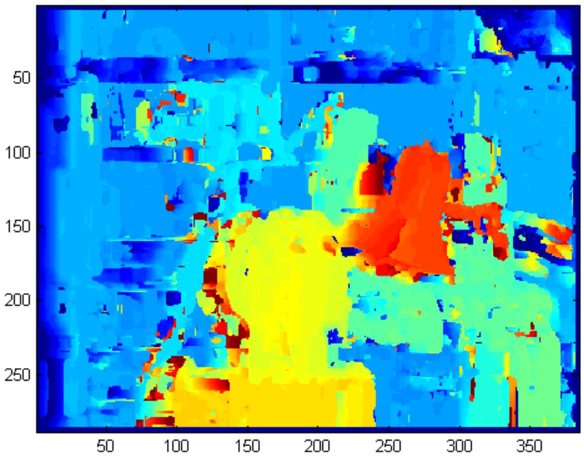
Di seguito viene mostrato lo schema a blocchi dell'algoritmo. L'algoritmo in MatLab, giustamente, non segue alla lettera lo schema poiché necessita delle opportune rettifiche nella traduzione in codice della procedura.

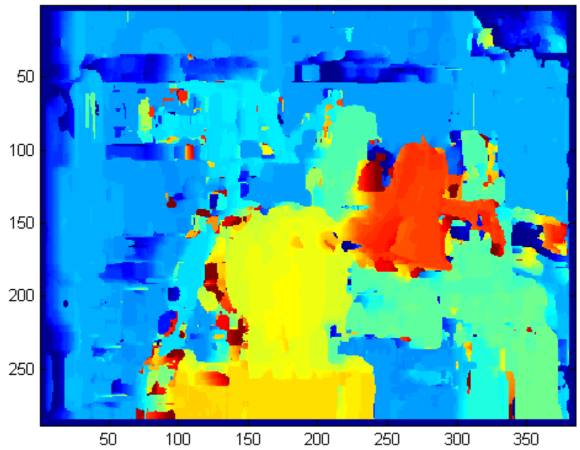


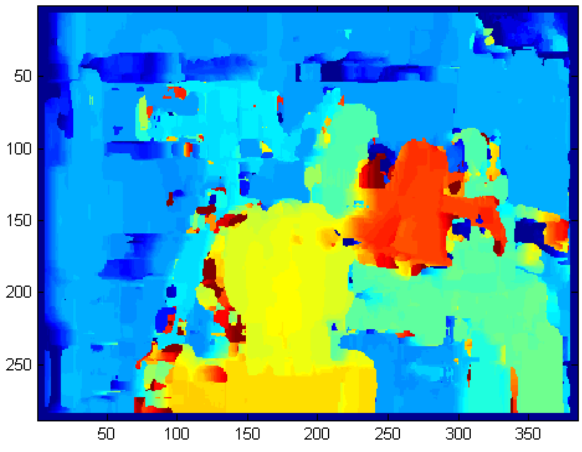
Di seguito vengono mostrati i risultati ottenuti con l’algoritmo tensor-oriented. In ordine, sinistra-destra su-giu, è stata applicata una finestra di controllo della disparità, 3x3, 5x5, 7x7, 9x9, 11x11











Bibliografia:

[1] - **Depth discontinuities pixel by pixel** - S. Birchfield, C. Tomasi: Stanford University, Computer Science Department.

[2] – **An Introduction to Computer Vision, Techniques & Algorithms** - B.Cyganek, J.P.Siebert: Wiley 2009, ISBN 978-0-470-01704-3.