http://www.dmi.unict.it/~iplab

Digital Video Stabilization

Image Processing Laboratory
Computer Science Department - University of Catania

Introduzione

Le videocamere portatili hanno riscontrato un successo crescente negli ultimi anni…

…ma la qualità dei filmati è bassa quando non viene utilizzato un treppiede o un punto di appoggio.

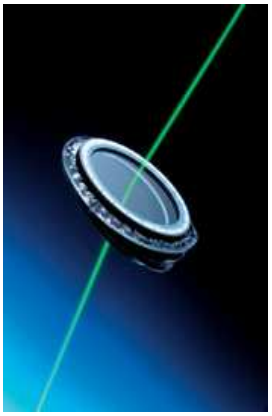
Video stabilizzazione

I sistemi di **video stabilizzazione** hanno l'obiettivo di eliminare gli effetti delle vibrazioni della videocamera, migliorando la qualità dei filmati.

Esistono diverse categorie di sistemi per la video stabilizzazione.

Stabilizzazione ottica

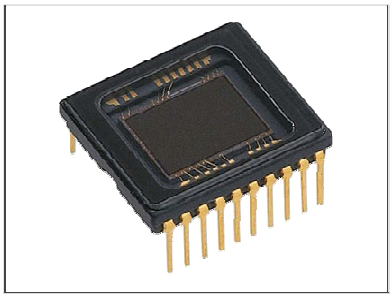
Tramite un gruppo ottico mobile la luce viene opportunamente deviata per annullare lo spostamento rilevato dai sensori.



- E' una tecnica molto accurata che riesce a correggere le vibrazioni in maniera istantanea;
- La qualità del filmato non viene intaccata, i fotogrammi vengono memorizzati già stabilizzati;
- Sfrutta dei componenti molto costosi.

Stabilizzazione elettronica

Ogni fotogramma viene stabilizzato modificando l'area sensibile del CCD, che viene spostata per compensare le vibrazioni rilevate.



- E' un sistema molto accurato, poiché il filmato viene memorizzato già stabilizzato;
- L'area del sensore effettivamente utilizzata è minore di quella disponibile;
- Impiega circuiti sofisticati e costosi.

Stabilizzazione meccanica

Utilizza dei giroscopi per eliminare le vibrazioni della video camera, sono stati creati anche dei sistemi a sospensione che agiscono da treppiede mobile.

- Gli apparati sono molto pesanti e ingombranti, è difficile usarli correttamente;
- Tali sistemi sono i meno adatti, di fatto limitano la libertà d'uso delle videocamere portatili.

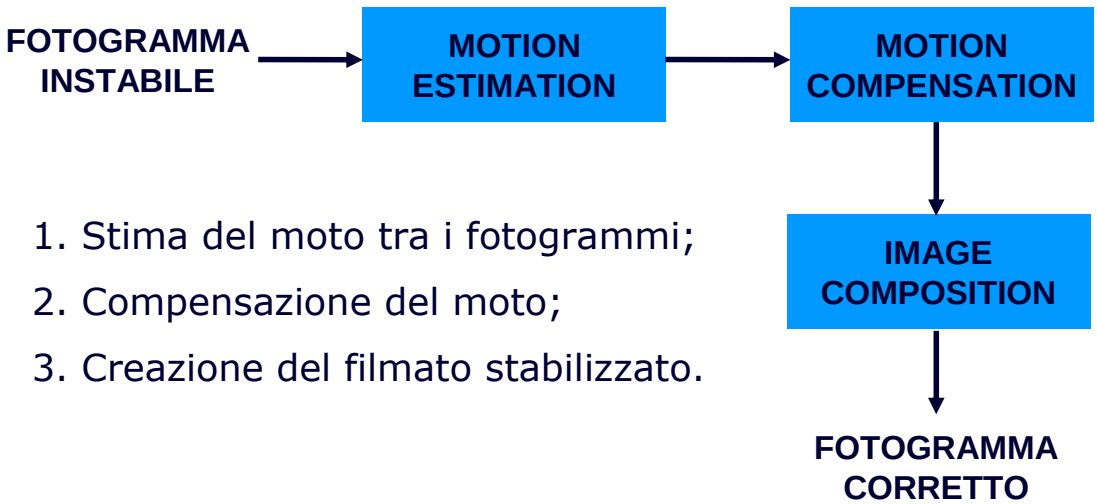
Stabilizzazione digitale

La **stabilizzazione digitale** utilizza solamente le informazioni contenute nel filmato, senza conoscere come si è mossa realmente la videocamera.

- Il moto della videocamera viene stimato dal contenuto della scena;
- Ogni fotogramma viene quindi ritagliato e riposizionato;
- Può avvenire in real-time o in post-processing;
- Molto semplice ed economica.

Stabilizzazione digitale (schema)

La generica architettura è divisa in tre unità funzionali che operano in sequenza su ogni fotogramma del filmato.



1. Stima del moto tra i fotogrammi;
2. Compensazione del moto;
3. Creazione del filmato stabilizzato.

Stabilizzazione Digitale (schema)

Il sistema *DIS* è generalmente costituito da tre unità:

- Motion estimation: valuta il movimento globale della telecamera tra immagini consecutive della sequenza e invia i parametri della trasformazione alla componente atta alla compensazione del moto;
- Motion compensation: effettua un'operazione di filtraggio distinguendo lo spostamento volontario della camera da quello involontario che dovrà essere corretto;
- Image composition: modifica l'immagine basandosi sui dati forniti dal MC e crea la sequenza stabilizzata.

Motion estimation

- Optical flow
- Block matching
- Feature based

Optical flow

- L'occhio umano percepisce il moto identificando punti corrispondenti in tempi differenti.
- La corispondenza è generalmente determinata assumendo che il colore o la luminosità di un punto non cambia durante il moto.
- Il moto 2D osservato o apparente, in computer vision, viene chiamato optical flow. L'optical flow può non coincidere con il reale moto 2D.

Esempi di flusso ottico

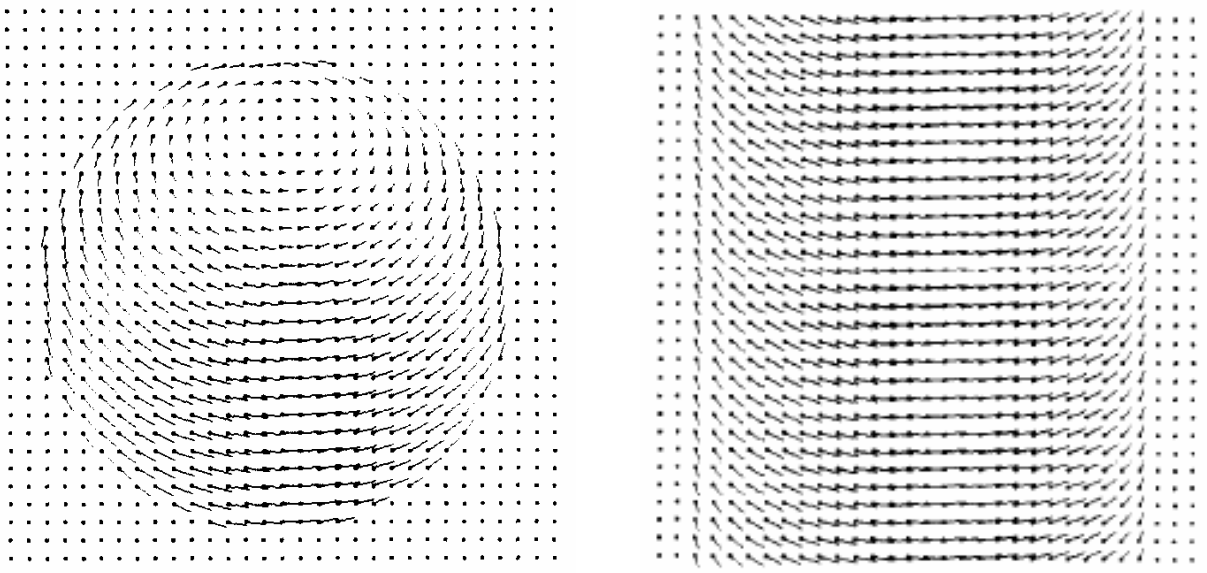


Fig. 2 Tratte da Determining Optical Flow, Berthold K.P. Horn and Brian G. Schunck.

Problemi del flusso ottico

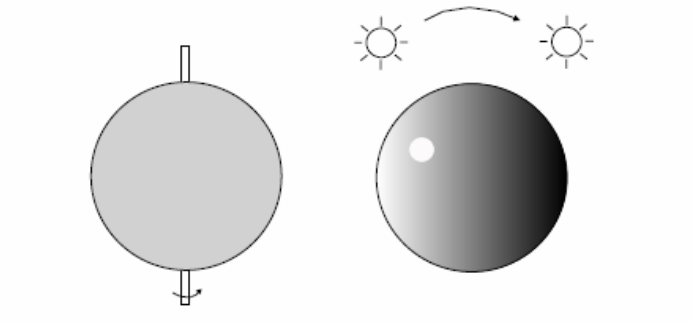


Fig. 3 Il flusso ottico e il campo di moto non sempre coincidono. In (a), una sfera sta ruotando in un ambiente ad illuminazione costante, ma l'immagine osservata non cambia. In (b), la sorgente luminosa sta ruotando attorno ad una sfera ferma, causando il movimento del punto luminoso sulla sfera. Tratte da Video Processing and Communication, Wang, Ostermann, Zhang.

Equazione del flusso ottico (1)

- Consideriamo una sequenza video la cui intensità luminosa è indicata dalla funzione I(x,y,t).
- Supponiamo che un punto (x,y) al tempo t si sia mosso in (x+dx,y+dy) al tempo t+dt. Sotto la condizione di intensità costante :

$$I(x+dx,y+dy,t+dt) = I(x,y,t)$$

Equazione del flusso ottico (2)

$$I(x, y, t) = I(x + dx, y + dy, t + dt)$$

Sviluppando in serie di Taylor il secondo membro e trascurando I termini di ordine superiore si ottiene:

$$I(x,y,t) = I(x,y,t) + \frac{\delta I}{\delta x}dx + \frac{\delta I}{\delta y}dy + \frac{\delta I}{\delta t}dt$$

Dunque:

$$\frac{\delta I}{\delta x}dx + \frac{\delta I}{\delta y}dy + \frac{\delta I}{\delta t}dt = 0$$

Equazione del flusso ottico (3)

$$\frac{\delta I}{\delta x}dx + \frac{\delta I}{\delta y}dy + \frac{\delta I}{\delta t}dt = 0$$

Dividendo ambo I membri per dt:

$$\frac{\delta I}{\delta x}v_x + \frac{\delta I}{\delta y}v_y + \frac{\delta I}{\delta t} = 0$$

Con v_x e v_y vettori velocità lungo l'asse x ed y.

Algoritmi Block based

- Partizioniamo dell'immagine in blocchi rettangolari;
- Supponiamo che l'intero blocco si muova solo in maniera traslazionale;
- Consideriamo una regione (scan area) nella quale cercare la nuova posizione del blocco;
- In base ad un *criterio di matching* tra il blocco stesso e l'area sottostante troviamo i parametri traslazionali.

Esempio Block matching

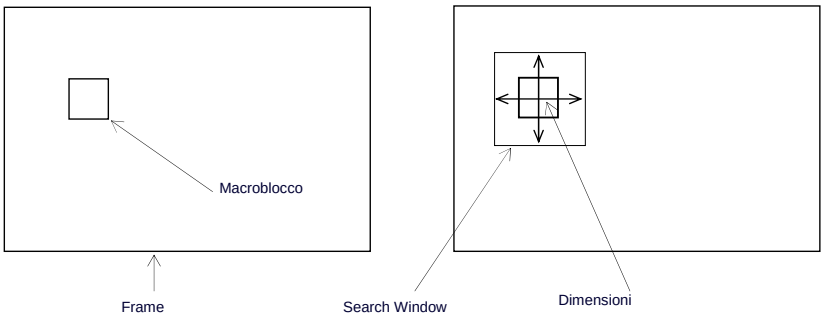
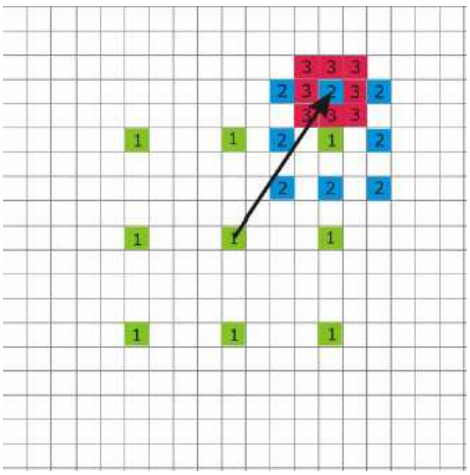


Fig.4 Esempio block matching.

Full search

- Immagine MxM, blocco NxN, range di ricerca R.
- Il numero di blocchi candidati per il matching è $(2R+1)^2$. Nel caso in cui si utilizzi la SAD come criterio di matching vengono effettuate un numero di operazioni proporzionali a N^2.
- Assumendo M multiplo di N ci sono $(M/N)^2$ blocchi nell'immagine.
- In definitiva occorre un numero di operazioni proporzionali a $M^2(2R+1)^2$. Considerando M=512, R=16 per ogni frame sono necessarie circa $2.85*10^8$ operazioni.

TSS (three step search)



1. Ad ogni passo TSS esamina il centro ed otto posizioni intorno ad esso. Il punto di distorsione minima diventa il centro del passo successivo.
2. La distanza iniziale delle otto posizioni da testare è metà della finestra di ricerca.
3. Ad ogni passo lo step viene dimezzato e si ferma quando arriva ad uno.

Il numero di operazioni per blocco è:

$$9 \cdot \left[\log_2(step_0)+1\right] \cdot SAD$$

Fig.5 Esempio di funzionamento di TSS

Confronto Optical flow-BMA

Block Matching	Gradient Based
Computazionalmente oneroso	Computazionalmente leggero
Adatto anche per grossi spostamenti	Adatto solo per piccoli spostamenti
Risoluzione fissa	Risoluzione variabile

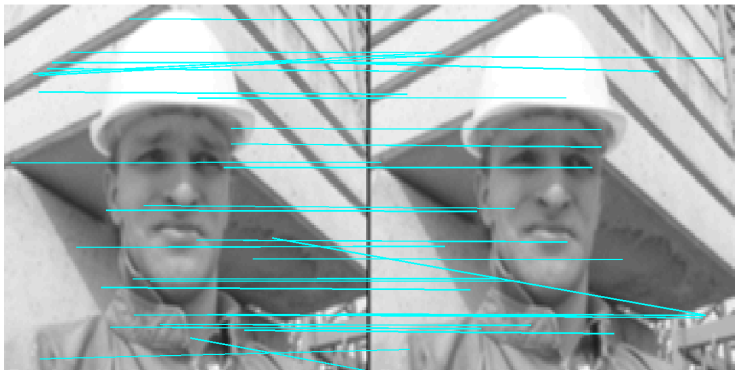
Vettori di moto

Vettori di moto filtrati (esempio 1)

Vettori di moto filtrati (esempio 2)

Feature

Una **feature** è una caratteristica di un'immagine che può essere automaticamente individuata: bordi, angoli, punti chiave, etc…



Vengono utilizzate da alcuni algoritmi per capire come si è mosso un fotogramma rispetto al precedente e stimarne così il vettore di moto

Feature (buone proprietà)

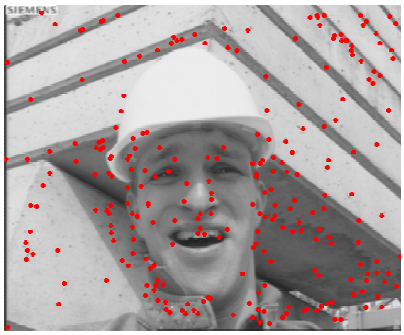
Le feature da utilizzare dovrebbero avere le seguenti caratteristiche:

- **Invarianza** rispetto a diverse trasformazioni dell'immagine (rotazione, scala, luminosità, etc.)
- **Presenza regolare** in immagini successive
- **Tracciabilità** in immagini diverse
- **Immunità** al rumore

Le feature SIFT presentano molte di queste qualità

SIFT (Scale Invariant Feature Transform)

Scale Invariant Feature Transform (SIFT) è un algoritmo progettato da David Lowe nel 2004 per estrarre feature significative da un'immagine.



Ad ogni punto individuato viene associato un descrittore che consente di individuarlo in immagini diverse.

SIFT Passi principali (1)

I passi principali dell'algoritmo di estrazione delle SIFT sono i seguenti:

- **Individuazione degli estremi locali nello scale-space:** si cercano punti interessanti su tutte le scale e posizioni dell'immagine utilizzando una funzione **DoG** (***Difference of Gaussian***). L'approccio utilizzato è quello del filtraggio in cascata (*cascade filtering approach*), che consente di determinare le posizioni e la scala delle feature candidate ad essere punti chiave e che, in un secondo momento, vengono caratterizzate con maggior dettaglio.
- **Localizzazione dei keypoint:** per ciascun punto candidato viene costruito un modello dettagliato per determinarne posizione e scala. I punti vengono inoltre selezionati secondo misure di stabilità.

SIFT Passi principali (2)

- **Generazione delle orientazioni:** per ottenere l'invarianza rotazionale, ad ogni punto chiave (*keypoint*) vengono associate una o più orientazioni calcolate in base ai gradienti locali dell'immagine.
- **Generazione del descrittore:** a partire dai gradienti locali dell'immagine, alla scala selezionata e nell'intorno del punto chiave, viene costruito il descrittore.

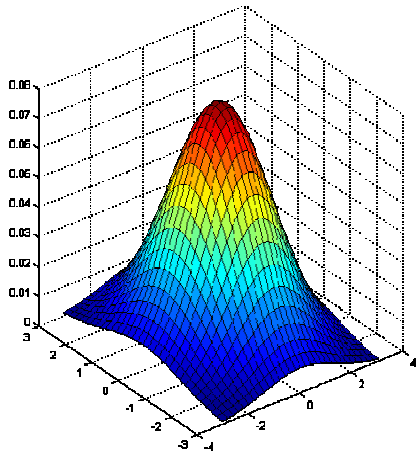
Analisi scale-space (1)

L'**analisi scale-space** di un'immagine è basata sul filtraggio progressivo con un filtro a scala variabile

Si utilizza un filtro gaussiano con sigma σ variabile che proietta l'immagine $I(x,y)$ in uno spazio tridimensionale $L(x,y,\sigma)$:

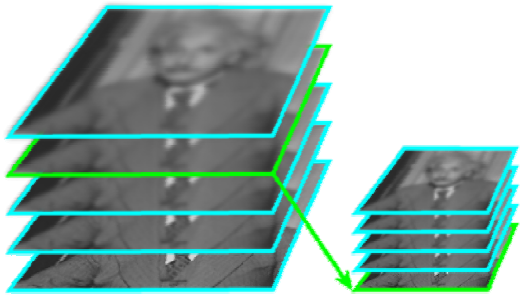
$$L(x,y,\sigma) = I(x,y) * G(x,y,\sigma)$$

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}}$$



Analisi scale-space (2)

Per migliorare le prestazioni l'immagine viene ridimensionata ad intervalli regolari e poi filtrata di nuovo, dando origine a gruppi di immagini detti **ottave**

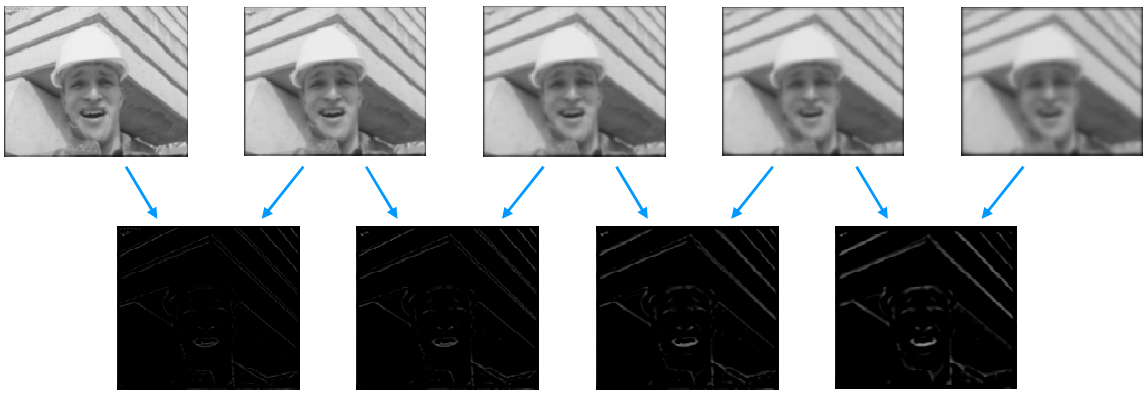


Per localizzare i punti chiave si utilizza quindi un filtro di *Differenze-di-Gaussiane,* separate da un fattore costante k:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma)$$

Analisi scale-space (3)

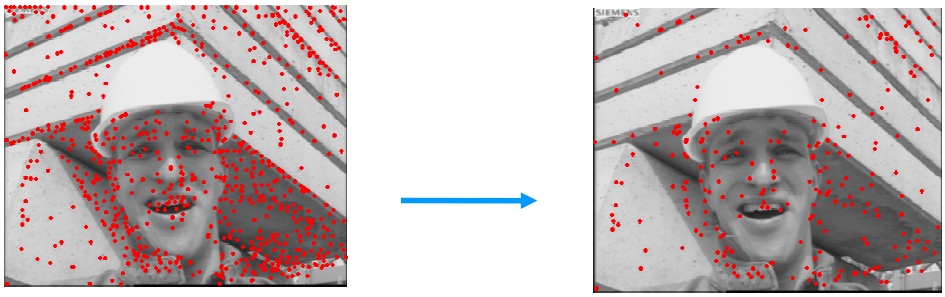
In ogni ottava le immagini adiacenti vengono sottratte a coppie, enfatizzando così i punti più significativi



Gli estremi relativi di queste immagini sono i **punti chiave**

Localizzazione dei keypoint

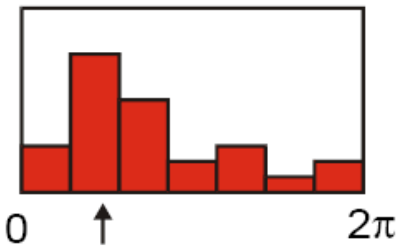
- Viene testata la stabilità dei punti chiave:
- analisi a soglia del contrasto
- verifica della presenza di bordi o linee
- interpolazione della posizione per aumentare la precisione della localizzazione



I punti che non superano tutti i test vengono scartati

Generazione delle orientazioni

Ad ogni punto chiave viene assegnata una **orientazione** che dipende dai gradienti nell'intera regione dell'immagine in cui esso si trova

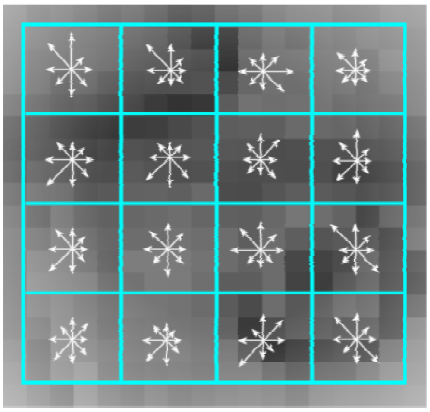


In questo modo si può ottenere l'invarianza del descrittore rispetto alle rotazioni

Generazione del descrittore

Ad ogni punto chiave viene associato un **descrittore** di 128 elementi, calcolato in base ai gradienti di tutti i pixel prossimi al punto chiave

L'algoritmo di costruzione del descrittore permette di ottenere l'invarianza dello stesso a numerose trasformazioni, tra cui di illuminazione, del punto di ripresa, della scala.



Keypoint matching (1)

In molte applicazioni (object recognition, stereo vision, motion analysis), occorre mettere in corrispondenza punti trovati in immagini diverse. Le sift trovano punti "interessanti" nelle varie immagini ed associano loro dei descrittori.

A partire dai descrittori è possibile mettere in relazione i punti trovati nelle varie immagini; dato un punto nell'immagine I_a con descrittore d_{pa} il punto corispondente nell'immagine I_b è quello che ha il descrittore più vicino (es. distanza euclidea) a d_{pa}.

Può accadere che non tutti i punti presenti in un'immagine hanno un corrispondente nell'altra immagine (occlusioni, nuovi oggetti nella scena …). Occorre dunque avere un criterio che permetta di capire se un punto ha un buon match.

L'utilizzo di una soglia fissa sulla distanza tra descrittori non si è dimostrata una buona scelta.

Keypoint matching (2)

Lowe utilizza come misura della bonta del matching la seguente relazione:

$$r = \frac{d1}{d2}$$

d1 distanza euclidea tra il descrittore d_{pa} ed il descrittore con minima distanza (d_{pb1} relativo ai punti dell'immagine I_b) da d_{pa}

d2 distanza euclidea tra il descrittore dpa ed il descrittore con minima distanza da d_{pa} dopo d_{pb1}

Nel caso di matching corretto r assumerà valori non elevati. In particolare se *r* è inferiore a 0.8 Lowe considera il matching è corretto, altrimenti il punto con descrittore d_{pa} non deve essere preso in considerazione.

SIFT (esempio)

Fig. Esempio di utilizzo delle SIFT. Tratta da: Distinctive Image Features from Scale-Invariant Keypoints, Lowe.

Motion compensation

La fase di motion compensation si occupa di riconoscere il movimento della camera voluto da quello non desiderato. I più noti approcci in letteratura sono:

- Integrated Motion Vector: utilizza la formula $V_{int}(n)=kV_{int}(n-1)+V_{act}(n)$, con *n* indice della frame, *k damping parameter* che controlla l'intensità della stabilizzazione, V_{int} Integrated Motion Vector e V_{act} attuale stima del moto tra il *frame* corrente e quello precedente.
- Frame position smoothing: effettua il filtraggio tramite un lowpass filter;
- Kalman filtering: Si modellizza il movimento volontario tramite un sistema dinamico lineare e viene utilizzata su di esso la teoria dei filtro di Kalman.

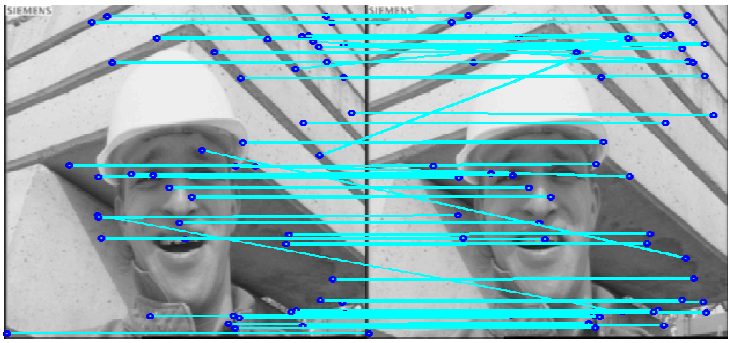
Approccio di video stabilizzazione basato sulle SIFT

L'algoritmo di video stabilizzazione è stato diviso in 3 parti:

1. Tracciamento delle feature tra fotogrammi consecutivi;
2. Stima del moto globale del fotogramma dallo spostamento che ogni feature ha subito;
3. Filtraggio delle eventuali componenti intenzionali del moto della videocamera per evitare correzioni errate.

Abbinamento feature

Dopo aver estratto le feature SIFT da due fotogrammi consecutivi viene fatto un abbinamento tra i due insiemi



Le feature vengono abbinate sfruttando la distanza euclidea tra i descrittori e un indice che viene calcolato per scartare con alta probabilità i ***"falsi positivi"***

Keypoint matching (1)

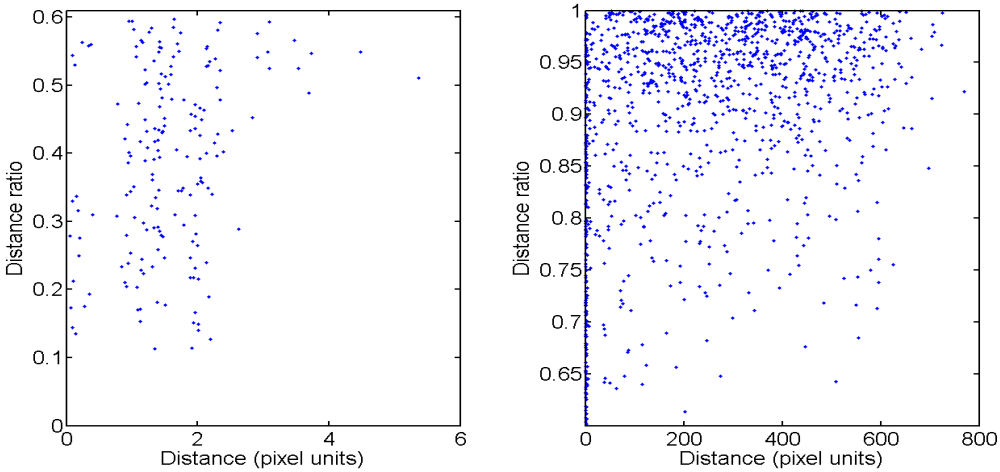
Durante l'abbinamento ogni descrittore viene confrontato con il gruppo di descrittori dell'altra immagine, trovando i due descrittori a distanza euclidea minima da esso:

$$r = \frac{d_1}{d_2} \qquad 0 \le r \le 1$$

Il rapporto tra la prima e la seconda distanza minima, detto **distance ratio**, è minore per gli abbinamenti corretti e maggiore per quelli errati

Keypoint matching (2)

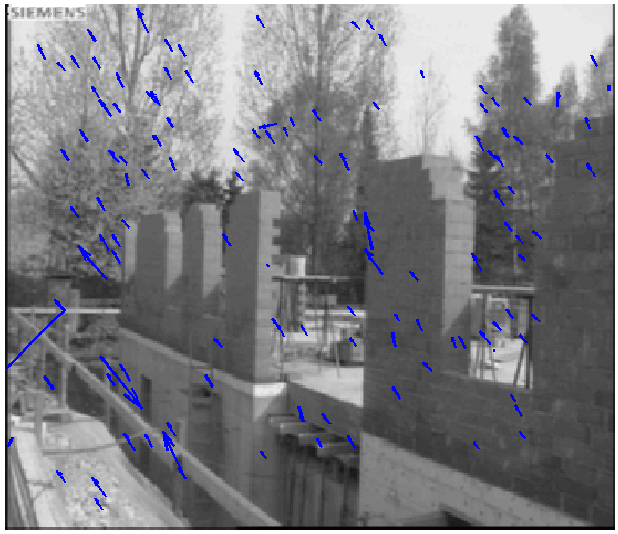
Questo indice è stato testato su coppie di fotogrammi, valutandone la relazione con la distanza euclidea tra i pixel abbinati



Indici maggiori di **0.6** corrispondono con buona probabilità ad abbinamenti errati

Stima del moto

Per ogni coppia di feature si calcola un **vettore di moto locale** che indica come si è spostato il punto: si ottiene quindi una prima stima del vettore di moto globale del fotogramma.



Movimenti della camera

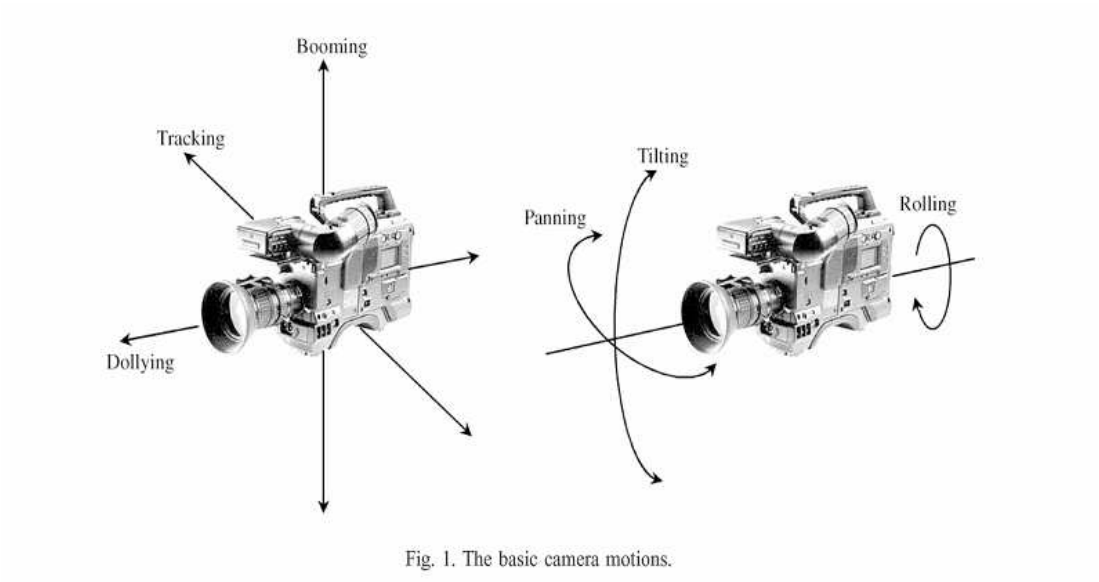


Fig. 1. The basic camera motions.

Qualitative estimation of camera motion parameters from the linear composition of optical flow", S. Park, H. Lee, S. Lee. *Pattern Recognition* 37 (2004) 767-779.

Modello utilizzato

Il movimento viene stimato con un modello affine bidimensionale che fornisce due traslazioni, un angolo di rotazione e un parametro di zoom:

$$\begin{cases} x_f = \lambda x_i \cos\theta - \lambda y_i \sin\theta + T_x \\ y_f = \lambda x_i \sin\theta + \lambda y_i \cos\theta + T_y \end{cases}$$

Poiché occorre stimare 4 parametri ma le coppie di punti presenti sono molto più numerose viene utilizzato il **metodo dei minimi quadrati**, in una sua versione modificata che riduce gli errori di stima.

Minimi quadrati (sensibilità agli outlier)

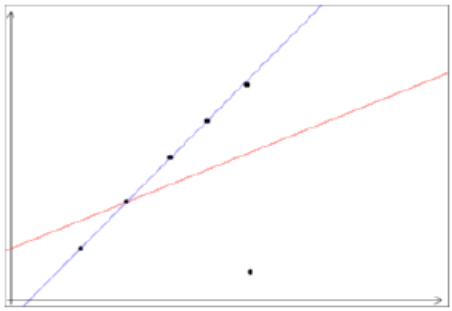
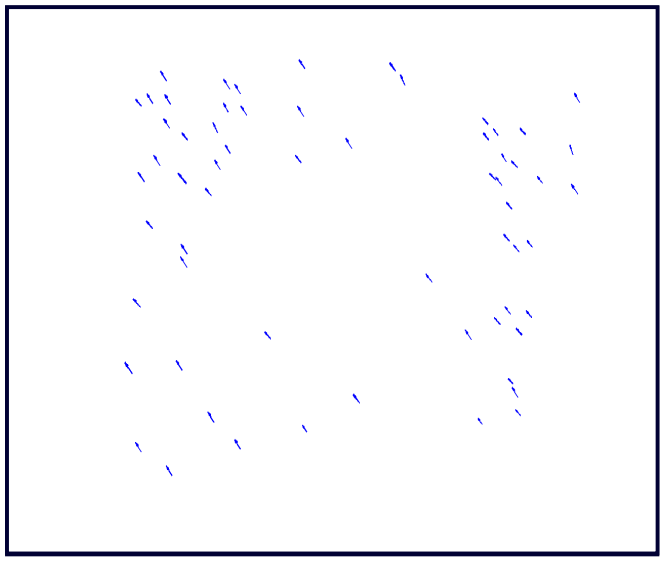


Figura A.3: Esempio di effetto di un outlier: retta di regressione ai minimi quadrati (in rosso) e retta di regressione robusta.

Minimi quadrati iterativi

I vettori locali forniscono una prima stima del moto globale: dopo aver calcolato delle misure d'errore si raffina la stima solo con i vettori locali corretti



Misure di errore

La prima stima effettuata consente di prevedere dove si dovrebbe trovare ogni feature nel fotogramma successivo.

Per poter scartare le feature errate vengono quindi calcolate due misure d'errore:

- **Distanza euclidea** tra il punto previsto e il punto trovato (sensibile ad errori sulla traslazione)
- **Angolo** compreso tra il punto previsto e il punto trovato, rispetto al centro dell'immagine (sensibile ad errori sulla rotazione)

Errore cumulativo

Quando una feature viene osservata attraverso fotogrammi consecutivi vengono cumulate entrambe le misure di errore :

$$E_N^C = (1-\alpha)E_{N-1}^C + \alpha E_N$$

$$\alpha = 0.35$$

Tale errore cumulativo viene utilizzato per scartare le feature che si discostano dalla stima: vengono mantenute soltanto le feature con gli errori più bassi e viene ripetuta la stima con esse

Questo metodo consente di utilizzare per la stima solo le feature rivelatesi più affidabili in passato

Filtraggio del moto intenzionale (1)

E’ necessario riconoscere il moto intenzionale della videocamera per non stabilizzarlo: viene utilizzata una tecnica adattativa di **Motion Vector Integration (MVI)** che riconosce l’eventuale movimento intenzionale

Si tratta di un meccanismo adattativo che osserva il movimento stimato nei fotogrammi passati per capire se sta avvenendo uno spostamento intenzionale, da non correggere, oppure uno spostamento casuale, da compensare.

Filtraggio del moto intenzionale (2)

Questa tecnica calcola il vettore di moto globale **GMV** del fotogramma corrente utilizzando un vettore di moto totale **IMV**, calcolato dopo ogni fotogramma:

$$IMV_N = \delta\, IMV_{N-1} + GMV_N$$

$$0 \le \delta \le 1$$

Così si ottiene la compensazione reale con cui correggere il fotogramma corrente:

$$C_N = IMV_N - IMV_{N-1}$$

Filtraggio del moto intenzionale (3)

La scelta del parametro di smorzamento consente di mantenere il moto intenzionale oppure correggere ogni movimento rilevato

La tecnica adattativa sceglie automaticamente il valore di tale parametro tra due valori possibili, dipendentemente dal moto rilevato nei fotogrammi precedenti:

- se si è osservato un movimento **regolare**, si sceglie lo **smorzamento maggiore;**
- se si è osservato un movimento **casuale**, si sceglie lo **smorzamento minore.**

Esempio (1)

Esempio (2)

Esempio (3)

Filmato de-stabilizzato

Filmato stabilizzato

Riferimenti

- K.P. Berthold Horn, B. G. Schunck. "Determining Optical Flow". Artificial Intelligence 17 pp. 185-203, 1981.
- D. Lowe - *Distinctive Image Features from Scale-Invariant Keypoints,* International Journal of Computer Vision Vol. 60(2), p.91 (2004).
- F. Vella, A. Castorina, M. Mancuso, G. Messina. "Digital Image Stabilization by Adaptive Block Motion Vector Filtering". IEEE Trans. on Consumer Electronics, Vol. 48, No. 3, pp. 796-801, August. 2002.
- J. Yang, D. Schonfeld, C.Chen, M. Mohamed - *Online Video Stabilization based on Particle Filters,* IEEE International Conference on Image Processing (2006).
- S. Auberger, C. Miro - *Digital Video Stabilization Architecture for Low Cost Devices,* Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis p. 474 (2005).
- S. Erturk. "Image Sequence Stabilization : Motion Vector Integration (MVI) Versus Frame Position Smoothing (FPS)". Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis, pp.266-271, 2001.
- S. Erturk, "Image sequence stabilization based on Kalman filtering of frame positions," Electronics Letters, vol. 37, no. 20, pp. 1217–1219, 2001.

Video Stabilization

Image Processing Laboratory
Computer Science Department - University of Catania

Q&A