

Chapter 1

Stato dell'arte

1.1 Speeded Up Robust Feature

L'algoritmo SURF (Speeded Up Robust Feature) è un descrittore e un detector robusto di caratteristiche locali di un'immagine. È diverse volte più veloce e robusto di SIFT, il descrittore a cui si ispira. SURF è stato presentato nel 2006 da Hebert Bay e ne sono state realizzate diverse implementazioni open source e commerciali: l'implementazione originale (commerciale) è scritta in c++, mentre *JavaSurf*, *JopenSURF*, *ImageJ SURF* e *BoofCV* sono implementazioni open source per Java.

Per cercare corrispondenze tra immagini in genere si procede con tre passi fondamentali:

1. Si individuano i punti di interesse (ad esempio i bordi) che hanno la caratteristica di ripetersi, ossia possono essere ritrovati da un detector anche se cambia la prospettiva di visualizzazione dell'immagine.
2. Si descrivono i punti di interesse tramite un vettore di caratteristiche. Questo vettore deve essere di grandi dimensioni se si intende privilegiare la robustezza dell'algoritmo oppure di piccole dimensioni se l'obiettivo è la velocità computazionale.
3. Si effettua il matching tra i vettori e le immagini.

SURF è nato con l'obiettivo di trovare un trade-off tra la velocità (Speeded-Up) e la robustezza (Robust). Per l'individuazione e la descrizione dei punti di interesse SURF utilizza l'approssimazione della matrice Hessiana, che è molto accurata, e le immagini integrali, che riducono drasticamente il tempo computazionale.

Un'immagine integrale $I_{\Sigma}(x, y)$ è la somma dell'intensità dei pixel compresi tra l'origine e il punto di coordinate (x, y) . In formule:

$$I_{\Sigma}(x, y) = \sum_{i=0}^x \sum_{j=0}^y I(i, j)$$

Ottenuta l'immagine integrale bastano quattro operazioni per calcolare un'area rettangolare di qualsiasi dimensione. Questa proprietà è sfruttata da SURF per la realizzazione di filtri di diverse dimensioni.

La matrice Hessiana permette di individuare le strutture blob (regioni in cui proprietà come la luminosità e il colore differiscono rispetto a quelle dell'ambiente) perchè in quei punti il determinante (o discriminante) è massimo, e di selezionare la scala dell'immagine; tuttavia le immagini con rotazioni di multipli dispari di $\frac{\pi}{4}$ perdono ripetibilità. Le risposte blob sono memorizzate in una mappa di risposte su scale differenti.

È fondamentale poter trovare i punti di interesse della stessa immagine su diverse scale poichè spesso si cercano corrispondenze tra due immagini rappresentate con scale diverse. Lo spazio delle scale può essere rappresentato come una piramide; con i metodi tradizionali (ad esempio SIFT) l'immagine viene continuamente ridimensionata mentre SURF ridimensiona solo il filtro: il calcolo delle scale può essere eseguito con attività in parallelo guadagnando velocità computazionale mentre con SIFT l'immagine ad ogni livello di scala dipende da quella precedente, per cui la computazione è sequenziale. La differenza tra i metodi SIFT e SURF è mostrata in Figura 1.1.

Figure/Figura1_1.png

Figure 1.1: A sinistra l'approccio SIFT (ridimensionamento dell'immagine), a destra il metodo SURF (ridimensionamento del filtro).

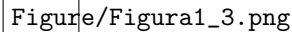
Il numero di punti di interesse individuati varia con la scala: chiaramente maggiore è la scala minore è il numero dei punti di interesse individuati, come mostrato nel seguente diagramma:

Figure/Figura1_2.png

Figure 1.2: Diagramma dei punti di interessi individuati in rapporto alla scala

I descrittori SURF descrivono la distribuzione dell'intensità del contenuto (pixel?) nell'intorno del punto di interesse in modo simile a SIFT. Si costruisce la distribuzione di primo ordine delle risposte della wavelet Haar nelle direzioni x e y e anche in questo caso si utilizzano le immagini integrali per aumentare la velocità.

Inizialmente si cerca un'orientazione riproducibile e indipendente dalla rotazione dell'immagine: la si ottiene dalle informazioni contenute in una regione circolare intorno al punto di interesse; in seguito si costruisce una regione quadrata allineata con l'orientamento trovato e si estrae il descrittore SURF composto da 64 componenti; infine si procede con il matching tra le immagini. Per quest'ultimo step si cerca di risparmiare tempo distinguendo i blob scuri da quelli chiari e confrontando solo i blob che hanno lo stesso tipo di contrasto.



Figure/Figura1_3.png

Figure 1.3: Due blob con contrasti differenti non sono presi in considerazione per il matching

1.2 Active Shape Models

I metodi basati su modello utilizzano una forma base che rappresenta l'immagine; questa viene utilizzata per trovare la miglior corrispondenza tra il modello e una nuova immagine. Questo approccio top-down è più semplice e meno soggetto a errori di quello bottom-up nel quale l'immagine viene analizzata per individuare particolari strutture e punti di interesse che la caratterizzano.

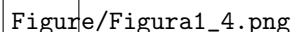
Gli Active shape models (ASMs) sono modelli statistici delle forme di oggetti che vengono iterativamente deformati fino ad adattarsi alle nuove immagini. Sono stati sviluppati da Tim Cootes e Chris Taylor nel 1995.

Il modello ASM è costruito in base all'analisi di un training set di immagini d'esempio e la forma dell'oggetto è rappresentato da un insieme di punti controllati dal modello.

L'algoritmo ASM alterna due passi fondamentali:

1. La generazione della forma base analizzando l'intorno di ogni punto per un miglior posizionamento del punto stesso.
2. L'adattamento della forma al modello di distribuzione dei punti.

Per il primo passo è necessario che l'algoritmo individui dei punti di riferimento sempre presenti nelle immagini appartenenti al training set: ciò implica che il training set deve contenere la stessa tipologia di oggetti e che questi ultimi abbiano una forma tale da prevedere punti di riferimenti ben distinti; ASMs non è adatto a immagini di oggetti amorfi.



Figure/Figura1_4.png

Figure 1.4: I punti di riferimento trovati nell'immagine di un volto