

# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Obiettivo/ Research question . . . . .	1
1.2	Stato dell'arte . . . . .	1
1.2.1	Speeded Up Robust Feature . . . . .	1
1.2.2	Active Shape Models . . . . .	4
1.2.3	Applicazioni simili . . . . .	6
	<b>Elenco delle figure</b>	<b>10</b>
	<b>Bibliografia</b>	<b>10</b>

# Capitolo 1

## Introduzione

*Se lo si desidera, utilizzare questo spazio per inserire un breve riassunto di ciò che verrà detto in questo capitolo. Inserire solo i punti salienti.*

### 1.1 Obiettivo/ Research question

Scrivere obiettivo e research question

### 1.2 Stato dell'arte

#### 1.2.1 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.

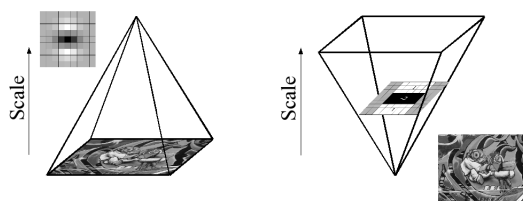


Figura 1.1: A sinistra l'approccio SIFT (ridimensionamento dell'immagine), a destra il metodo SURF (ridimensionamento del filtro).

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:

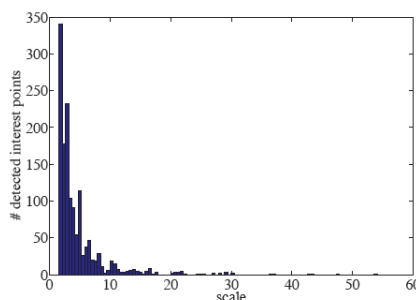


Figura 1.2: Diagramma dei punti di interessi individuati in rapporto alla scala

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.



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

### 1.2.2 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.

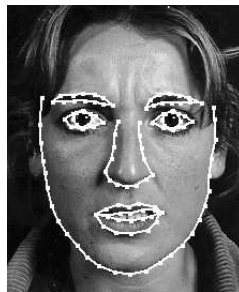


Figura 1.4: I punti di riferimento trovati nell'immagine di un volto

ASM è conveniente da utilizzare nel caso in cui:

- Si possiede un insieme di immagini rappresentative del soggetto da rappresentare;
- Gli oggetti hanno una forma definita;
- Si conosce con buona approssimazione il target dell'immagine.

Nel caso invece di immagini in cui gli oggetti possono assumere forme molto diverse è preferibile utilizzare un approccio “bottom-up” (estrazione delle features).

### **1.2.3 Applicazioni simili**

Molto spesso capita che ad un museo o ad una mostra vorremmo approfondire le informazioni dell'autore o dell'opera che stiamo osservando. Tale motivazione unita al progresso tecnologico ha portato allo sviluppo di applicazioni che permettono di far ciò semplicemente facendo una fotografia.

#### **Artfinder**

Artfinder è un'applicazione per Iphone o Ipad che permette, scattando una foto ad un quadro, scultura o disegno, di avere tutte le informazioni che si desiderano. Se la ricerca da esito negativo è possibile aggiungere l'opera con i relativi dati. Inoltre ha anche funzioni social: offre l'opportunità di condividere le passioni artistiche con amici e follower. Artfinder non è solo un riconoscitore di immagini, ma permette di riconoscere dove sono in mostra gli artisti preferiti, di avere una preview delle opere esposte e di creare un'art list. Si possono consultare, inoltre, orari delle esposizioni e, grazie a un sistema di geolocalizzazione, segnalare tutti i musei, le gallerie o le mostre che sono facilmente raggiungibili dal luogo in cui ci si trova. Se non si è in possesso di uno smartphone o un tablet Apple, Artfinder è anche un sito internet sul quale si possono ricercare i quadri, acquistare copie direttamente online oppure creare la tua galleria virtuale.

## Google Goggles

Google Goggles è un'applicazione per il riconoscimento visivo delle immagini con tecnologia OCR<sup>1</sup>. Tale applicazione ha molteplici possibilità di utilizzo, rilevamente quella di riconoscimento delle opere d'arte, la quale è valsa a Google un accordo con il Metropolitan Museum of Art per fornire link diretti al proprio sito sui capolavori in esso esposti. Inoltre ha anche le seguenti funzionalità:

- trova informazioni scattando una foto su punti di riferimento, libri, negozi, quadri etc;
- scannerizzando un barcode fornisce dettagli sul prodotto associato ad esso;
- esegue la scansione dei biglietti da visita e comprende i dati pertinenti per crearne un contatto;
- risolve i sudoku.

Tutte le attività riportate sopra si basano sulla possibilità di riconoscere gli oggetti o il testo nell'immagine catturata dal telefono. Esso utilizza tecniche multiple per il riconoscimento delle immagini: in primo luogo si cerca di identificare l'oggetto con alcuni algoritmi di riconoscimento e lo confronta con delle immagini di un database di Google. Per aiutare la ricerca si cerca di trovare del testo nell'immagine utilizzando il riconoscimento ottico dei caratteri per avere un'idea migliore di ciò che l'oggetto potrebbe essere. Utilizza anche il GPS per capire dove si trova l'utente per filtrare i risultati ricevuti per i luoghi di interesse con quelli che sono rilevanti dalla posizione. Ci sono diversi algoritmi per l'Object Recognition:

---

<sup>1</sup>Optical Character Recognition, sono programmi dedicati alla conversione di un'immagine contenente testo.



- **Edge Detection**, i contorni in un'immagine di solito sono robusti al cambiamento di illuminazione/colore. L'esecuzione di algoritmi di rilevamento sull'immagine, come Canny Edge Detection<sup>2</sup>, sono in grado di rilevarli nel modello e nell'immagine. In seguito vengono confrontati con le possibili soluzioni del modello.
- **Scale Invariant Feature Transform**, i punti chiave delle immagini sono estratti prima da un insieme di immagini di riferimento e poi archiviate in un database. Un oggetto è riconosciuto in una nuova immagine comparando individualmente ogni feature di questa con il database e cercare candidati che corrispondono alle feature basate sulla distanza euclidea fra i loro vettori di feature.

Google Goggles era inizialmente così sensibile che in molti casi poteva trovare una persona attraverso un'immagine e restituire il link ad un blog che conduce a lui. Google si rese conto dei problemi di privacy e ora controlla i risultati in modo da non riconoscere le persone. Essendo ancora in fase di sperimentazione non tutti i risultati sono accurati, infatti non lavora bene con mobili e vestiario. Ci sono comunque diverse accorgimenti per ottenere risultati migliori con Google Goggles:

- scattare foto in ambienti con buona illuminazione;
- zoommare ciò che si vuole fotografare;
- utilizzare il pulsante di ritaglio per concentrarsi sull'area d'interesse;
- usare il telefono con orientamento 'paesaggio';

---

<sup>2</sup>Algoritmo per il riconoscimento dei contorni, utilizza un metodo di calcolo multi-stadio per individuare contorni di molti dei tipi normalmente presenti nelle immagini reali.

- tenere le mani ferme ed utilizzare sullo schermo il pulsante di scatto.

# Elenco delle figure

1.1	A sinistra l'approccio SIFT (ridimensionamento dell'immagine), a destra il metodo SURF (ridimensionamento del filtro). . . . .	3
1.2	Diagramma dei punti di interessi individuati in rapporto alla scala . .	3
1.3	Due blob con contrasti differenti non sono presi in considerazione per il matching . . . . .	4
1.4	I punti di riferimento trovati nell'immagine di un volto . . . . .	5

# Bibliografia

- [1] C. Bonivento - C. Melchiorri - R. Zanasi, “*Sistemi di controllo digitale*”, Progetto Leonardo, 1995.
- [2] H. Bay - A. Ess - T. Tuytelaars - L. Van Gool, “*Speeded-Up Robust Features (SURF)*”, 2008.
- [3] <http://en.wikipedia.org/wiki/SURF>, “*SURF*”, Wikipedia.
- [4] T. Cootes, “*An Introduction to Active Shape Models*”, 2008.
- [5] [http://en.wikipedia.org/wiki/Active\\_shape\\_model](http://en.wikipedia.org/wiki/Active_shape_model), “*Active shape model*”, Wikipedia.