

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA  
Facoltà di Scienze Matematiche, Fisiche e Naturali  
Corso di Laurea Magistrale in Scienze e Tecnologie Informatiche  
(Polo Scientifico - Didattico di Cesena)

---

Tesi di Laurea

**TECNICHE DI RICONOSCIMENTO  
DEL VOLTO DA IDENTIKIT**

Relatore:  
Prof.ssa Annalisa Franco

Candidato:  
Simone Buoncompagni

---

Anno Accademico 2011-2012

Ai miei genitori

# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Biometria e riconoscimento del volto . . . . .	1
1.2	Riconoscimento da sketch/identikit . . . . .	3
1.3	Struttura della tesi . . . . .	4
<b>2</b>	<b>Riconoscimento del volto da sketch</b>	<b>6</b>
2.1	Stato dell'arte . . . . .	6
2.1.1	Approcci generativi . . . . .	7
2.1.2	Approcci discriminativi . . . . .	10
<b>3</b>	<b>Contributo del lavoro</b>	<b>13</b>
3.1	Presentazione del sistema complessivo . . . . .	13
3.2	Pre - processing del volto . . . . .	14
3.3	Riconoscimento mediante estrazione di feature . . . . .	19
3.3.1	Feature di forma . . . . .	19
3.3.2	SURF (Speeded - Up Robust Features) . . . . .	21
<b>4</b>	<b>Pre - processing del volto</b>	<b>25</b>
4.1	Estrazione dei contorni: IVP Framework . . . . .	25
4.2	Normalizzazione: BioLab - ICAO Framework . . . . .	31
<b>5</b>	<b>Calcolo delle feature e riconoscimento</b>	<b>35</b>
5.1	Feature di forma . . . . .	35
5.1.1	Gray Level Profile . . . . .	36
5.1.2	Shape Matrix . . . . .	41
5.1.3	Local Orientation Histogram . . . . .	44
5.1.4	Fourier Descriptors . . . . .	49
5.1.5	Local Binary Pattern . . . . .	55
5.1.6	Beam Angle Statistics . . . . .	58
5.1.7	Chain Code e Pixel Decimal Value . . . . .	62
5.2	SURF (Speeded - Up Robust Features) . . . . .	64
5.3	Riconoscimento mediante fusione dei risultati . . . . .	70

<b>6</b>	<b>Implementazione</b>	<b>74</b>
6.1	Piattaforma di sviluppo e librerie utilizzate . . . . .	74
6.2	Organizzazione del prototipo software . . . . .	75
<b>7</b>	<b>Prove sperimentali</b>	<b>78</b>
7.1	Organizzazione delle prove sperimentali . . . . .	78
7.2	Training del sistema . . . . .	80
7.3	Testing del sistema . . . . .	86
<b>8</b>	<b>Conclusioni e sviluppi futuri</b>	<b>89</b>
8.1	Riepilogo dello studio effettuato . . . . .	89
8.2	Risultati ottenuti . . . . .	90
8.3	Possibili sviluppi futuri . . . . .	91
	<b>Bibliografia</b>	<b>92</b>
	<b>Ringraziamenti</b>	<b>97</b>

# Elenco delle figure

1.1	Esempio di software per la creazione di identikit . . . . .	3
2.1	Confronto tra l'approccio basato su <i>eigentransformation</i> [7] (prima riga) e l'approccio basato su separazione di texture e forma (seconda riga) [8]: (a) fotografia in input, (b) sketch ottenuto con il processo di sintesi proposto, (c) sketch disegnato dall'artista	8
2.2	Risultati del processo di sintesi in pseudo sketch: (a) foto del volto, (b) sketch disegnato dall'artista, (c) sketch ottenuto con l'approccio proposto da Liu <i>et al.</i> [9], (d) sketch ottenuto con il metodo <i>eigentransformation</i> [7] . . . . .	8
2.3	Approccio di Li <i>et al.</i> [12]: nella prima colonna la foto originale, nella seconda lo sketch, nella terza la foto ricostruita a partire dallo sketch . . . . .	9
2.4	Conversione da foto a sketch proposta Wang e Tang [13]: (a) foto, (b) sketch disegnato dall'artista, (c) sketch ottenuto dal processo di conversione . . . . .	9
2.5	Esempi di coppie foto - sketch somiglianti in base all'approccio proposto da Yuen e Man [15]: (a) coppie con similarità bassa, (b) coppie con similarità media, (c) coppie con similarità elevata	10
2.6	Una panoramica della fase di training (a) e della fase di matching diretto (b) costituenti il cosiddetto LFDA framework proposto da B. Klare, A. Jain e Z. Li. [20] . . . . .	11
3.1	Schema a blocchi del sistema di riconoscimento . . . . .	14
3.2	Schema a blocchi riassuntivo della fase di pre - processing con input costituito da uno sketch del volto . . . . .	15
3.3	Comparazione del risultato di estrazione dei contorni a partire da uno sketch e da una foto afferenti al medesimo soggetto . . .	17
3.4	Comparazione del risultato di estrazione dei contorni a partire da uno sketch e da una foto afferenti a soggetti diversi . . . . .	18
3.5	Schema a blocchi riassuntivo del calcolo di una feature di forma	21
3.6	Schema a blocchi riassuntivo relativo al calcolo della feature SURF	23
3.7	Schema a blocchi riassuntivo relativo all'individuazione delle fotografie più somiglianti ad un dato sketch . . . . .	24

4.1	Interfaccia grafica del framework IVP e generico esempio di rete funzionale per l'estrazione dei contorni di un'immagine . . . . .	27
4.2	Composizione della rete funzionale per l'estrazione dei contorni da foto e sketch . . . . .	27
4.3	Risultati intermedi degli step di estrazione dei contorni a partire da una fotografia . . . . .	30
4.4	Risultati intermedi degli step di estrazione dei contorni a partire da uno sketch . . . . .	30
4.5	Esempio di un volto con un buon allineamento (a sinistra) e di un volto che necessita di essere allineato (a destra) . . . . .	31
4.6	Esempio di angolo che sussiste tra l'asse che congiunge i due centri degli occhi e l'asse orizzontale . . . . .	32
4.7	Risultato della rilevazione del bounding box del volto . . . . .	33
4.8	Risultato della rilevazione dei centri degli occhi . . . . .	33
4.9	Immagine contenente i contorni del volto prima della normalizzazione (a sinistra) e dopo la normalizzazione (a destra) . . . . .	34
5.1	Bounding box calcolato a partire dall'immagine dei contorni del volto: l'area inclusa all'interno sarà considerata per l'estrazione delle feature . . . . .	36
5.2	Gray Level Profile: la linea blu rappresenta la divisione in due metà del volto, mentre su ognuna delle linee rosse orizzontali si effettua il conteggio dei pixel di foreground . . . . .	38
5.3	Gray Level Profile: istogramma risultante dal processo di conteggio dei pixel di foreground (sopra) e corrispondente istogramma normalizzato a bin raggruppati (sotto) . . . . .	39
5.4	Gray Level Profile: istogrammi (in rosso quelli relativi alle parti sinistre dei due volti, in verde quelli relativi alle parti destre) calcolati da uno sketch e da una foto afferenti al medesimo soggetto . . . . .	39
5.5	Gray Level Profile: istogrammi (in rosso quelli relativi alle parti sinistre dei due volti, in verde quelli relativi alle parti destre) calcolati da uno sketch e da una foto afferenti a soggetti distinti . . . . .	40
5.6	Shape Matrix: griglia a 15 righe e 10 colonne sovrapposta all'immagine dei contorni del volto per il calcolo della feature . . . . .	41
5.7	Shape Matrix: rappresentazione grafica dei pesi applicati alle celle della griglia (a colore più scuro corrisponde peso più basso) . . . . .	43
5.8	Local Orientation Histogram: risultato del calcolo dell'immagine direzionale a partire da due generiche immagini . . . . .	45
5.9	Local Orientation Histogram: immagine direzionale ottenuta a partire dall'immagine dei contorni. Su quest'ultima viene sovrapposta una matrice $M \times N$ (in rosso) e l'immagine direzionale complessiva è ottenuta come composizione delle immagini direzionali calcolate su ogni cella separatamente dalle altre . . . . .	47

5.10	Fourier Descriptors: esempio della funzione centroid distance calcolata su una generica forma . . . . .	50
5.11	Fourier Descriptors: funzione centroid distance calcolata su 3 forme distinte ma simili . . . . .	51
5.12	Fourier Descriptors: ricostruzione delle 3 forme di Figura 5.11 a partire dai primi 3 descrittori di Fourier (a) e a partire dai primi 5 descrittori di Fourier (b) ottenuti dal processo di trasformata delle rispettive funzioni centroid distance . . . . .	51
5.13	Fourier Descriptors: diversi casi di ricostruzione di una forma a partire dai primi $M$ descrittori di Fourier . . . . .	52
5.14	Fourier Descriptors: campionamento dei punti esterni di bordo del volto per la costruzione della shape signature . . . . .	53
5.15	Fourier Descriptors: descrittori alle basse frequenze per uno sketch ed una foto relativi al medesimo soggetto . . . . .	54
5.16	Fourier Descriptors: descrittori alle basse frequenze per uno sketch ed una foto relativi a soggetti distinti . . . . .	54
5.17	Local Binary Pattern: calcolo dell'operatore LBP su un'immagine grayscale del volto del soggetto . . . . .	56
5.18	Local Binary Pattern: intorno circolare di un pixel per il calcolo di LBP . . . . .	56
5.19	Local Binary Pattern: esempi di strutture di pattern uniformi .	57
5.20	Local Binary Pattern: pattern uniformi e non uniformi individuati all'interno di un'immagine grayscale del volto . . . . .	57
5.21	Local Binary Pattern: istogrammi costituenti il vettore di feature	58
5.22	Beam Angle Statistics: esempio di calcolo dell'angolo di ordine $k$ (con $k = 5$ ) . . . . .	59
5.23	Beam Angle Statistics: contorno originale (a), contorno rumoroso (b), plot dei momenti BAS di ordine 1 (c), 2 (d), 3 (e) . . . . .	60
5.24	Beam Angle Statistics: punti campione (crocette blu) e linea di delimitazione (linea rossa) per il calcolo della feature sul contorno esterno del volto . . . . .	61
5.25	Chain Code: semplice esempio di calcolo delle etichette numeriche associate ai vertici dei pixel . . . . .	62
5.26	Pixel Decimal Value: maschera $3 \times 3$ (a) da scorrere sull'immagine per il calcolo dei valori decimali (b) . . . . .	63
5.27	SURF: esempio di calcolo di una porzione dell'immagine integrale	65
5.28	SURF: analisi a differenti valori di scala . . . . .	67
5.29	SURF: calcolo dell'orientazione dominante . . . . .	67
5.30	SURF: esempi di finestre quadrate a dimensione e orientazione variabile centrate sui punti d'interesse individuati . . . . .	68
5.31	SURF: suddivisione della finestra quadrata in sottoregioni e calcolo del descrittore a 64 componenti . . . . .	68

5.32	Calcolo dei punti SURF sulle immagini originali (normalizzate ed allineate) di foto e sketch . . . . .	69
6.1	Screenshot dell'applicazione prototipo per il riconoscimento sketch - foto . . . . .	75
6.2	Screenshot dell'applicazione prototipo per la parametrizzazione di una delle feature di forma . . . . .	76
6.3	Screenshot dell'applicazione prototipo in caso d'utilizzo di una sola feature per il riconoscimento . . . . .	77
6.4	Screenshot dell'applicazione prototipo in caso d'utilizzo di più feature per il riconoscimento . . . . .	77
7.1	Dataset CUHK: esempi di volti contenuti nel dataset . . . . .	79
7.2	Dataset AR: esempi di volti contenuti nel dataset . . . . .	79
7.3	Training Set CUHK: confronto tra le percentuali di Rank - $N$ Recognition Rate in caso d'adozione di un sistema di punteggio a scarto unitario e non unitario . . . . .	85
7.4	Testing Set CUHK: confronto tra le percentuali di Rank - 5 Recognition Rate in caso d'adozione di un sistema di punteggio a scarto unitario e non unitario . . . . .	87
7.5	Testing Set AR: confronto tra le percentuali di Rank - $N$ Recognition Rate in caso d'adozione di un sistema di punteggio a scarto unitario e non unitario . . . . .	88

# Elenco delle tabelle

5.1	Assegnamento dei punteggi alle fotografie ai primi 15 posti e ai primi 10 posti . . . . .	73
7.1	Risultati ottenuti sul training - set CUHK (88 soggetti) utilizzando singolarmente le diverse feature . . . . .	84
7.2	Alcune combinazioni di feature che consentono di ottenere i migliori risultati in termini di accuratezza di riconoscimento sul dataset di training; la percentuale indicata tra parentesi è la percentuale di Rank - 5 Recognition Rate, mentre l'altra è quella di Rank - 1 Recognition Rate . . . . .	84

# Capitolo 1

## Introduzione

### 1.1 Biometria e riconoscimento del volto

Il termine biometria deriva dalle parole greche *bios* (vita) e *metros* (misura) e si riferisce allo studio e all'impiego di metodi per rilevare e misurare caratteristiche di organismi viventi e trarne comparativamente classificazioni e leggi. Le sue innumerevoli applicazioni trovano riscontri nella biologia, in medicina, in genetica, nelle scienze agrarie e forestali, nelle scienze ambientali e in altri settori affini.

Nell'ambito informatico il termine biometria fa esplicito riferimento all'identificazione automatica o alla verifica dell'identità di una persona sulla base di caratteristiche biologiche; conseguentemente prende il nome di sistema biometrico quel sistema in grado di assolvere a tale compito.

Le caratteristiche biometriche che vengono sfruttate per il riconoscimento possono essere svariate e danno origine a diverse tipologie di sistemi di riconoscimento:

- sistemi basati sul riconoscimento di aspetti statici (o caratteristiche fisiologiche), cioè sistemi che operano una valutazione di caratteristiche fisiologiche dell'individuo quali impronta digitale, volto, mano, iride, retina, orecchio, dna, ecc.;
- sistemi basati sul riconoscimento di aspetti dinamici, cioè che operano una valutazione di caratteristiche comportamentali quali andatura, voce, firma, stile di battitura, ecc.;
- sistemi basati sul riconoscimento di aspetti chimico - fisici od organici, cioè che operano una valutazione di proprietà quali odore, presenza di virus e di anticorpi.

Nel contesto di questa tesi, la caratteristica biometrica oggetto di studio è la caratteristica statica per eccellenza: il volto. Il riconoscimento del volto, infatti, è una delle tecniche più utilizzate dagli esseri umani nella loro interazione

visuale e la sua acquisizione è solitamente un'operazione non intrusiva. Inoltre, il volto è l'unico elemento sfruttabile per il riconoscimento in tutti quei casi nei quali non è possibile avere collaborazione da parte del soggetto.

I contesti nei quali il volto viene utilizzato per il riconoscimento sono svariati e consistono in videosorveglianza, riconoscimento dei soggetti in aeroporti e passaggi di frontiera (recenti direttive ISO/ICAO prevedono l'uso del volto come caratteristica biometrica principale per i passaporti) e autenticazione nei sistemi di accesso.

In tutti questi casi il problema del riconoscimento è tutt'altro che di facile risoluzione visto che il volto può essere interessato da:

- invecchiamento;
- diverse espressioni facciali;
- variazioni dell'illuminazione e dello sfondo dell'ambiente circostante;
- variazioni di posizioni rispetto alla telecamera;
- presenza di occhiali;
- occlusioni di parti del volto a causa dei capelli.

Queste problematiche, che sono causa di una elevata variabilità intraclassa, si sommano al fatto che i volti umani sono contraddistinti da una elevata similarità interclasse. Per tali motivi, negli ultimi trent'anni la ricerca si è prodigata al fine di dar vita a sistemi di riconoscimento automatici il più possibile affidabili e robusti rispetto alle problematiche indicate in precedenza.

Nonostante il volto si sia rivelato una caratteristica biometrica poco adatta per applicazioni che richiedono un elevato grado di sicurezza come ad esempio i sistemi di accesso e sebbene per tali applicazioni il volto abbia assunto una posizione di sfavore nei confronti di caratteristiche biometriche considerate più affidabili (impronte digitali, iride), negli ultimi anni sono emerse le potenzialità del volto come strumento di ausilio per le forze dell'ordine e per l'ambito forense. In particolare, il riconoscimento di un soggetto a partire da un *identikit* o a partire da un generico disegno (*sketch*) del proprio volto è un problema molto interessante dal punto di vista delle possibili applicazioni pratiche.

Oggigiorno questo tipo di riconoscimento viene svolto da persone incaricate che si assumono il compito di confrontare visivamente il disegno del volto del soggetto e le foto segnaletiche a disposizione. Come vedremo meglio nel prosieguo, l'obiettivo di questo progetto di tesi è lo studio di tecniche automatiche innovative per il riconoscimento da sketch, basate sull'estrazione di caratteristiche che accomunino fotografie e sketch e ne rendano possibile un confronto diretto.

## 1.2 Riconoscimento da sketch/identikit

La vera identità di un individuo è un'informazione molto preziosa. In generale le persone non hanno particolari remore a rivelare la propria identità e il riconoscimento non rappresenta un problema; l'unica motivazione che può portare un individuo a nascondere la propria identità è l'elusione dei rilevamenti delle forze dell'ordine per un certo tipo di attività criminale. In questi casi, risulta difficile procedere al riconoscimento attraverso caratteristiche biometriche come iride o impronte digitali, in quanto la loro acquisizione presuppone un certo grado di collaborazione da parte del soggetto. Il volto, invece, può essere ottenuto in modo segreto, non solo attraverso le ben note telecamere di videosorveglianza ma anche attraverso la ricostruzione fornita da un eventuale testimone oculare che assiste all'evento criminoso. Quest'ultimo, attraverso la descrizione verbale del sospettato e con l'ausilio delle capacità di un disegnatore, può fornire una rappresentazione del volto del soggetto aiutando quindi il processo di individuazione dell'identità del malvivente. Questa rappresentazione prende solitamente il nome di identikit o sketch: a livello tecnico i due termini sono differenti, in quanto col termine sketch si denota un generico disegno mentre col termine identikit si intende il sistema specifico d'identificazione personale usato dalla polizia. Nel prosieguo di questo documento ci riferiremo al disegno parlando di 'sketch', in quanto il nostro vero interesse è su ciò che il disegno rappresenta, piuttosto che sul come o in che contesto esso viene ottenuto. Inoltre, è bene puntualizzare che gli sketch del volto possono essere ottenuti non solo dall'estro creativo di un artista, ma anche tramite appositi software. Queste applicazioni, come si può notare a titolo esemplificativo in Figura 1.1, mettono a disposizione delle 'primitive' del volto (sopracciglia, bocca, naso, varie forme del volto, ecc.) che possono essere selezionate singolarmente fino a comporre il viso nel suo complesso. La creazione del volto mediante questi

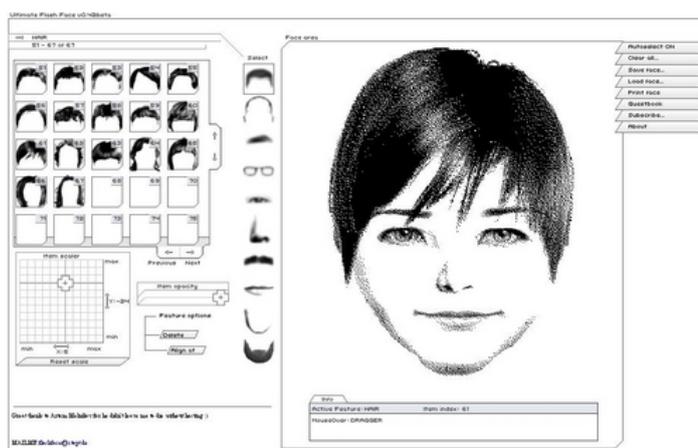


Figura 1.1: Esempio di software per la creazione di identikit

strumenti consente di semplificare la successiva fase di riconoscimento; molto più complesso è il caso del riconoscimento trattato in questa tesi, visto che verranno considerati come casi di studio i disegni effettuati a penna o a matita da appositi artisti.

Solitamente, una volta prodotto lo sketch, si effettua un confronto visivo con le fotografie segnaletiche già in possesso degli inquirenti e memorizzate all'interno di un'ipotetica banca dati. Tale confronto viene eseguito 'manualmente' da persone incaricate, comportando un inevitabile spreco di tempo; dotarsi di un sistema automatico in grado di ricercare la foto del sospettato più somigliante allo sketch a partire da quelle presenti all'interno di un database potrebbe aiutare le forze di polizia in una più celere individuazione dei sospettati. Inoltre, un ipotetico sistema di ricerca potrebbe aiutare il testimone e l'artista stesso a modificare lo sketch in modo interattivo sulla base delle fotografie restituite come risultato dall'operazione di ricerca.

Purtroppo, i metodi presenti allo stato dell'arte sul riconoscimento facciale non consentono un confronto diretto tra disegni e fotografie principalmente per due motivi:

- fotografia e disegno sono due rappresentazioni del volto molto differenti in termini di ricchezza di dettagli e tessitura e pertanto non direttamente confrontabili con tecniche standard;
- sono perlopiù ignoti i meccanismi psicologici che possono portare il testimone o il disegnatore ad enfatizzare alcuni aspetti del volto piuttosto che altri, influenzando quindi in modo determinante il processo di riconoscimento.

Come vedremo nel successivo capitolo dedicato allo stato dell'arte, numerosi gruppi di ricerca si sono dedicati nell'ultimo decennio allo studio di questa interessante tematica, proponendo diverse tecniche ed algoritmi in grado di valutare il grado di somiglianza tra fotografie e disegni dei volti e consentendo di oltrepassare la barriera rappresentata dalle due differenti modalità di rappresentazione.

L'obiettivo del progetto di tesi descritto in questo documento è quello di esplorare una nuova strategia di valutazione della similarità tra uno sketch e una fotografia consistente principalmente in un'analisi della conformazione dei volti, mediante estrazione dei contorni e calcolo di apposite feature.

### 1.3 Struttura della tesi

La tesi è organizzata in un totale di 8 capitoli. In questo primo capitolo è stato collocato il lavoro di tesi nel relativo ambito scientifico di riferimento. Nel Capitolo 2 viene fornita una descrizione dello stato dell'arte per quanto concerne il riconoscimento del volto da sketch, mentre nel Capitolo 3 viene illustrata a grandi linee la soluzione proposta e realizzata per questo tipo di problematica.

Nel successivo Capitolo 4 viene descritta la fase di pre - processing delle immagini dei volti, mentre nel Capitolo 5 sono illustrate le feature e il loro impiego per il riconoscimento. Nel Capitolo 6 viene brevemente indicato l'ambiente di implementazione e le scelte adottate per la produzione del sistema software complessivo. Nel Capitolo 7 vengono descritte le prove sperimentali fornendo anche i risultati di questi test e, infine, nel Capitolo 8 si indicano conclusioni e possibili sviluppi futuri.

## Capitolo 2

# Riconoscimento del volto da sketch

### 2.1 Stato dell'arte

Come anticipato nel precedente capitolo introduttivo, sulla tematica del riconoscimento del volto la ricerca si è adoperata notevolmente nel corso degli ultimi 20/30 anni. Gli algoritmi studiati e sviluppati sia nell'ambito 2D sia considerando modelli 3D hanno consentito di ottenere prestazioni sempre migliori in relazioni alle difficoltà endemiche del riconoscimento del volto, come le già accennate variazioni di luminosità e rotazione del volto, varietà delle espressioni facciali, invecchiamento e così via. Pur nella loro diversità, gran parte degli algoritmi sviluppati presentano un punto di partenza comune: si basano su tecniche *photo - based*, cioè tecniche che prevedono elaborazioni esclusivamente a partire dalla medesima 'materia prima' come appunto le immagini fotografiche, ottenute presumibilmente in condizioni differenti e con strumenti differenti ma pur sempre della stessa tipologia.

Considerando invece il riconoscimento da sketch e le possibili applicazioni si presentano delle problematiche completamente diverse da quelle descritte in precedenza e gli algoritmi *photo - based* si rivelano inefficaci. Variazioni dell'espressione, della posizione del volto o della luminosità sono assai rare viste le condizioni controllate nelle quali si effettuano l'acquisizione della fotografia del volto e la produzione del disegno. A costituire il problema principale è invece la diversa modalità di rappresentazione del volto stesso, visto che da un lato si ha un'immagine reale e dall'altro un'immagine disegnata, il che comporta differenze a livello di texture e di forma. Il 'lavoro' di una matita su un foglio di carta produce un effetto molto diverso se comparato all'aspetto reale del viso raffigurato in una foto; inoltre, in un disegno possono essere enfatizzate alcune caratteristiche facciali come se si trattasse di una sorta di caricatura e questo induce la presenza nel disegno di marcate deformazioni rispetto alla figura reale.

Per tutti questi motivi, la ricerca sulla tematica del riconoscimento sketch - foto si è sviluppata in due direzioni parallele:

- comprensione dei meccanismi psicologici che regolano e influenzano la produzione di uno sketch;
- individuazione di tecniche per confrontare tra loro immagini reali e disegni.

Per quanto concerne il primo punto, sono state effettuate diverse analisi e ricerche in materia di psicologia che hanno fatto da precorritrici ad una tematica che sarebbe stata affrontata a livello tecnico anni più avanti; è possibile citare [1], [2], [3], [4], [5]. Questi contributi, che per motivi di opportunità non approfondiamo ulteriormente, si sono concentrati sulla sorprendente capacità dell'essere umano di riconoscere oggetti e volti anche di fronte a caricature o storpiature delle forme, permettendo di intuire che è proprio sulle caratteristiche caricaturizzate che il cervello umano basa il riconoscimento.

A livello algoritmico, che è indubbiamente quello sul quale ci interessa maggiormente soffermarci, gli studi sul riconoscimento da sketch si sono mossi seguendo due strade alternative [6]:

- approccio generativo: consiste nel tentare di modellare o convertire un'immagine reale del volto in una rappresentazione sotto forma di sketch in modo da poter effettuare un confronto diretto con il disegno a disposizione, o viceversa nel tentare di ottenere una foto a partire da uno sketch;
- approccio discriminativo: consiste nell'individuazione ed estrazione di possibili feature invarianti rispetto alla differente modalità di rappresentazione del volto senza procedere ad alcuna conversione preliminare.

Lo stato dell'arte prevede tecniche e algoritmi che ricadono nell'una o nell'altra categoria e nel prosieguo di questo capitolo vedremo più nel dettaglio che cosa la ricerca ha prodotto nell'ultima decade per entrambi gli approcci.

### 2.1.1 Approcci generativi

Gli approcci generativi, come già anticipato, prevedono una trasformazione preliminare della foto in uno sketch (o viceversa); in questo modo è possibile effettuare un confronto diretto tra il risultato della conversione e l'altra immagine a disposizione mediante l'utilizzo delle tecniche di face recognition allo stato dell'arte. Wang e Tang hanno ad esempio proposto [7] l'utilizzo della trasformata KLT (*Karhunen-Loeve Transform*) sia per la conversione preliminare da foto a sketch sia per la successiva fase di *matching* tra sketch, dando quindi vita ad un algoritmo basato sul calcolo di particolari autovettori chiamati *eigensketch*. In un altro approccio [8], i medesimi ricercatori hanno presentato un algoritmo in grado di separare le informazioni legate a forma e texture

della foto ed ottenere conseguentemente uno sketch (Figura 2.1), demandando poi l'operazione di confronto e riconoscimento ad un classificatore Bayesiano. Importante citare anche il contributo di Liu *et al.* [9] i quali hanno proposto un approccio di sintesi da foto a sketch che preserva la geometria del volto (Figura 2.2) e che si basa su un classificatore discriminativo non - lineare.

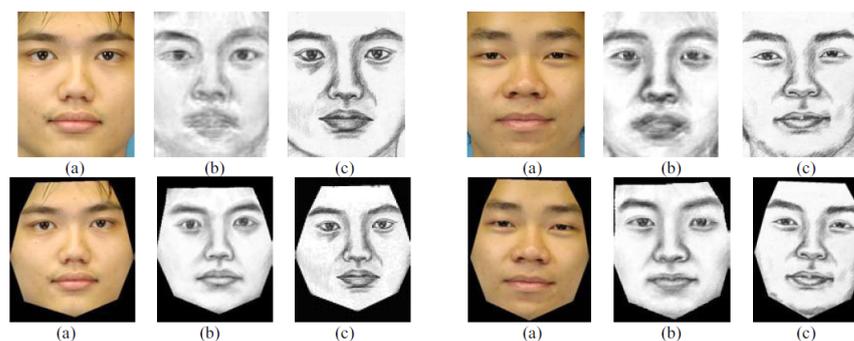


Figura 2.1: Confronto tra l'approccio basato su *eigentransformation* [7] (prima riga) e l'approccio basato su separazione di texture e forma (seconda riga) [8]: (a) fotografia in input, (b) sketch ottenuto con il processo di sintesi proposto, (c) sketch disegnato dall'artista

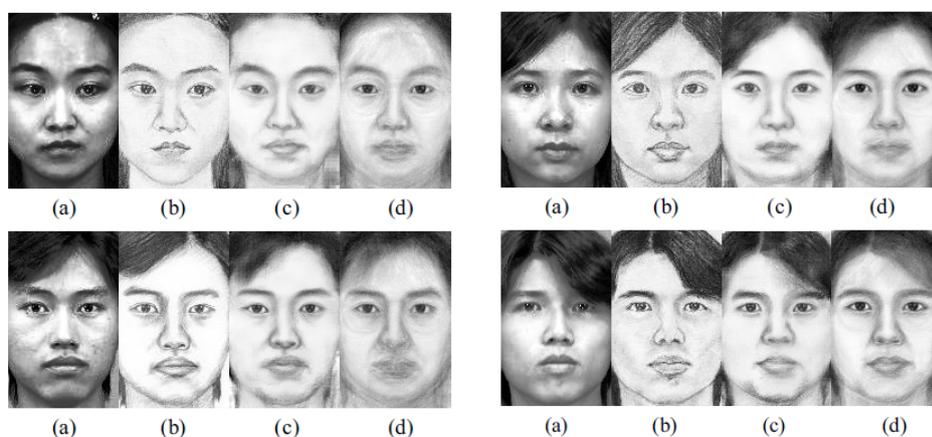


Figura 2.2: Risultati del processo di sintesi in pseudo sketch: (a) foto del volto, (b) sketch disegnato dall'artista, (c) sketch ottenuto con l'approccio proposto da Liu *et al.* [9], (d) sketch ottenuto con il metodo *eigentransformation* [7]

Tu e Lien [11] propongono un modello alternativo di conversione da foto a sketch chiamato DCM (*Directed Combined Model*) il quale si basa su informazioni della texture del volto oltre che della forma, mentre Sun *et al.* [10] propongono la costruzione di un *Active Shape Model* che consenta di effettuare la costruzione dello sketch a partire da alcuni particolari feature point presenti

nella foto originale del volto del soggetto.

Come il lettore avrà sicuramente notato, gli approcci fino ad ora citati prevedono un passaggio da foto a sketch ma esistono anche studi in senso opposto. Ad esempio, l'algoritmo di sintesi di Li *et al.* [12] è in grado di ottenere un'immagine realistica del volto del soggetto a partire da uno sketch mediante un complesso metodo basato su sottospazi ibridi (Figura 2.3). Il successivo confronto tra foto avviene invece mediante un filtro di correlazione invariante rispetto alle variazioni di illuminazione. Infine, a conclusione di questo paragrafo, è bene citare un recente approccio proposto da Wang e Tang [13] basato sull'utilizzo di un modello MRF (*Markov Random Fields*) in grado di eseguire automaticamente la conversione da sketch a foto (Figura 2.4) e viceversa.

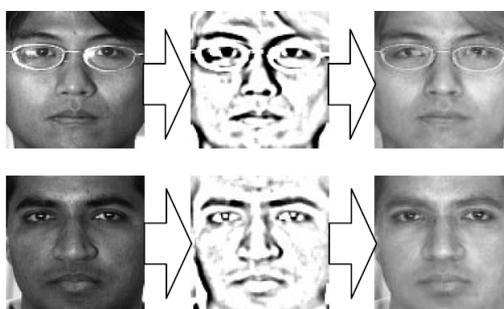


Figura 2.3: Approccio di Li *et al.* [12]: nella prima colonna la foto originale, nella seconda lo sketch, nella terza la foto ricostruita a partire dallo sketch

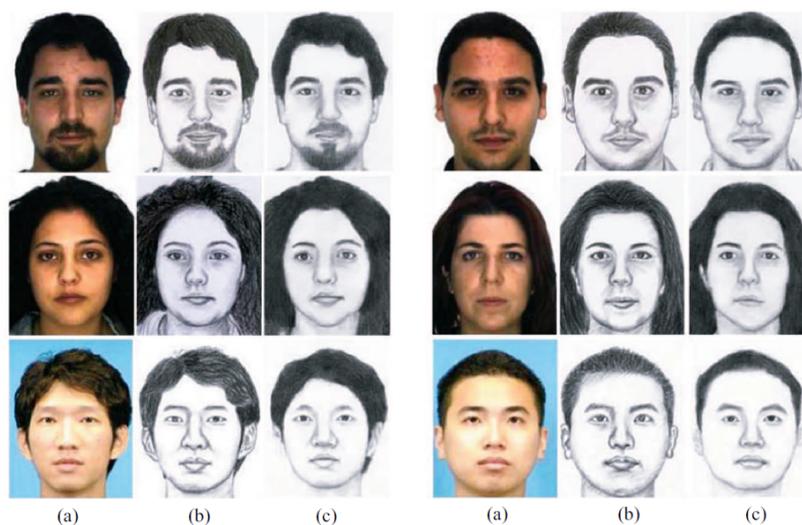


Figura 2.4: Conversione da foto a sketch proposta Wang e Tang [13]: (a) foto, (b) sketch disegnato dall'artista, (c) sketch ottenuto dal processo di conversione

### 2.1.2 Approcci discriminativi

Gli approcci discriminativi a differenza di quelli generativi tentano di confrontare direttamente sketch e immagini reali del volto senza procedere ad alcuna conversione preliminare. Come visto per le tecniche generative, anche quelle discriminative sono state oggetto di studi e ricerche e numerosi sono i contributi che possono essere citati. Uhl e Lobo [14], ad esempio, hanno proposto una standardizzazione fotometrica degli sketch in modo da poterli confrontare con le normali fotografie, per poi eseguire una normalizzazione geometrica dei due tipi di immagine e procedere al confronto mediante riduzione di dimensionalità e uso di autovalori/autovettori. Yuen e Man [15] propongono invece un sistema di matching basato su feature locali e globali estratte dalle due tipologie di immagini ed un sistema di feedback per migliorare il meccanismo di ricerca della foto più simile ad un dato sketch; in Figura 2.5 possiamo vedere alcuni esempi di coppie foto - sketch organizzate per grado di somiglianza rilevato dall'approccio proposto.

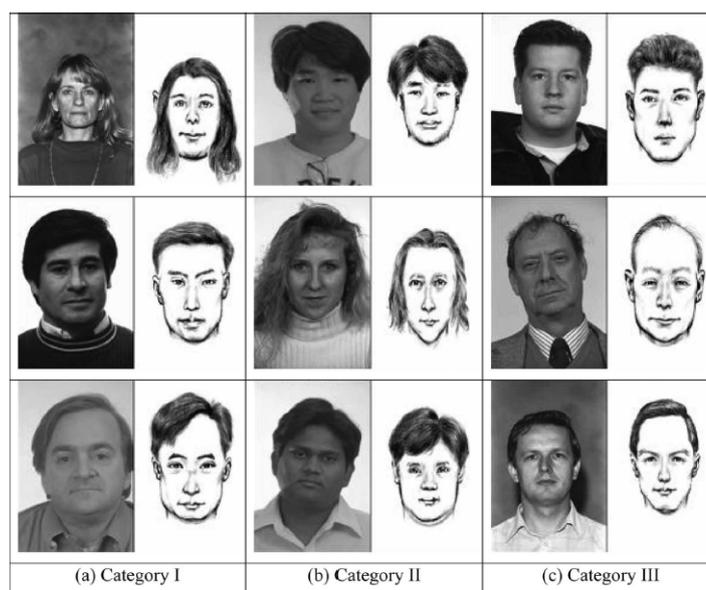


Figura 2.5: Esempi di coppie foto - sketch somiglianti in base all'approccio proposto da Yuen e Man [15]: (a) coppie con similarità bassa, (b) coppie con similarità media, (c) coppie con similarità elevata

Zhang *et al.* [16] nel loro studio effettuano un confronto tra le performance di riconoscimento sketch - foto da parte degli esseri umani e da parte di algoritmi basati sulla cosiddetta *Principal Component Analysis*, considerando anche variazioni di sesso, età, etnia e artista deputato alla redazione del disegno. Viene valutata inoltre la qualità degli sketch in termini di capacità dell'artista, esperienza, tempo di esposizione e dinstintività delle feature [17]. In modo simile,

Nizami *et al.* [18] valutano le differenze di prestazione nel matching con sketch disegnati da diversi artisti.

Importante sottolineare i diversi contributi di B. F. Klare e A. K. Jain [19], i quali hanno proposto un metodo di confronto basato sull'uso dell'operatore SIFT (*Scale Invariant Feature Transform*) in grado di individuare punti d'interesse all'interno di entrambe le tipologie di immagine a partire da un'analisi del valore e dell'orientazione del gradiente all'interno di regioni locali. I medesimi due ricercatori con la collaborazione di Z. Li [20] migliorano i risultati ottenuti creando un framework di lavoro chiamato LFDA (*Local Feature Discriminant Analysis*); esso prevede, prima del matching diretto, una fase di training del sistema che fa' uso di una versione multiscala della feature *Local Binary Pattern* e della già menzionata SIFT (Figura 2.6). Inoltre, in un loro recente studio [21], Klare e Jain propongono un ulteriore framework per effettuare un riconoscimento da sketch in cui le immagini reali del volto possono provenire, oltre che dai consueti dispositivi di acquisizione, anche da apparecchiature fotografiche a raggi infrarossi.

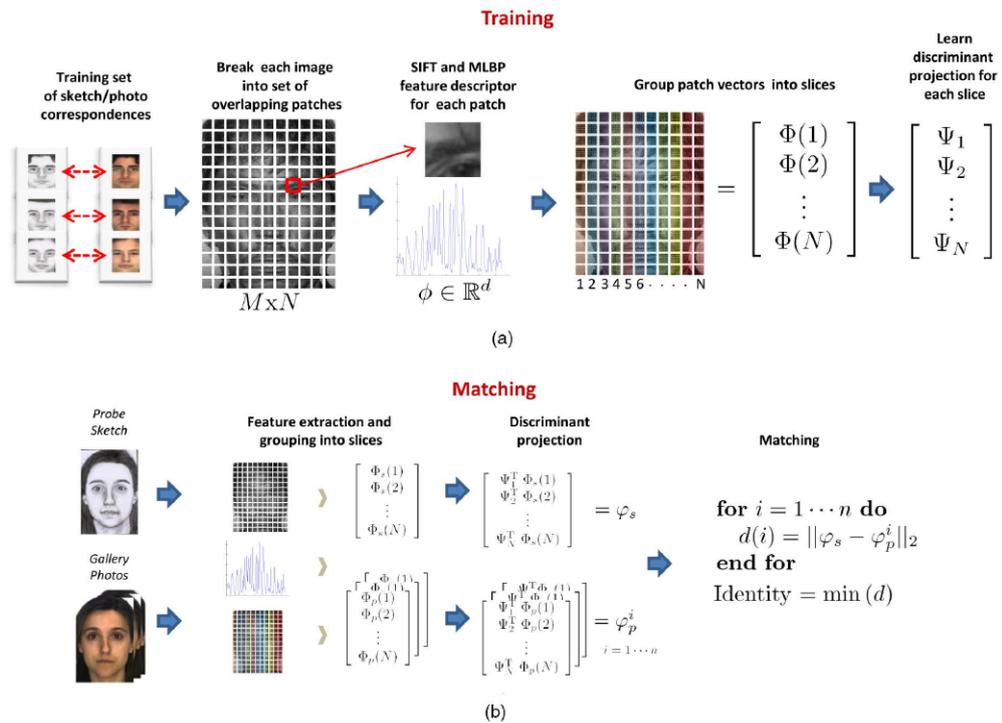


Figura 2.6: Una panoramica della fase di training (a) e della fase di matching diretto (b) costituenti il cosiddetto LFDA framework proposto da B. Klare, A. Jain e Z. Li. [20]

A conclusione di questa panoramica, è opportuno menzionare il contributo di Bhatt *et al.* [22] i quali estendono il cosiddetto *Uniform Local Binary Pattern*

in modo tale da poter lavorare con differenti feature a livello di texture su sketch e su foto e l'approccio di Zhang *et al.* [23] consistente in una nuova tipologia di descrittore del volto con permettete di considerare allo stesso modo sia gli sketch che le foto.

## Capitolo 3

# Contributo del lavoro

### 3.1 Presentazione del sistema complessivo

Dopo aver introdotto la tematica del riconoscimento del volto da sketch e averne mostrato lo stato dell'arte in questo capitolo sarà descritta ad alto livello la soluzione studiata in questa tesi per il problema in esame. Come si è visto nel precedente capitolo, ad oggi sono presenti in letteratura due categorie di approcci, uno generativo ed uno discriminativo. La tecnica analizzata e implementata sotto forma di software si colloca nell'ambito degli approcci discriminativi; l'obiettivo è infatti quello di dar vita ad un sistema di riconoscimento del volto da sketch che non preveda alcun tipo di conversione preliminare da foto a sketch (o viceversa) ma che esegua il riconoscimento confrontando direttamente il disegno e l'immagine reale del volto del soggetto. Inoltre, il caso di studio considerato è costituito dallo scenario tipico che si può presentare nel momento in cui occorre individuare l'identità di un sospettato; il sistema realizzato è cioè strutturato per lavorare in modalità di 'identificazione' ed effettuare quindi confronti uno a molti. Più precisamente, come mostrato in Figura 3.1, l'input è rappresentato da uno sketch e da un insieme di foto, mentre l'output è costituito da una classifica delle fotografie più simili allo sketch; questo è utile non solo per valutare l'effettiva bontà della tecnica di riconoscimento studiata e implementata, ma anche per permettere ad ipotetici utilizzatori del sistema di avere a disposizione una pluralità di soggetti potenzialmente somiglianti a quello ricercato. La tecnica proposta per il riconoscimento e che in quanto tale costituisce il cuore di funzionamento dell'applicazione implementata si basa sull'intuizione di trattare i volti come se fossero delle forme geometriche 'speciali' e, in quanto tali, analizzabili attraverso il calcolo di particolari feature di forma. Queste feature, che saranno illustrate in dettaglio nelle successive sezioni di questa relazione, forniscono sia una descrizione di alcune caratteristiche salienti del volto sia un valido strumento di comparazione tra sketch e fotografie. È importante sottolineare come il calcolo di questi particolari descrittori non venga effettuato direttamente sulle foto e sugli sketch, bensì su

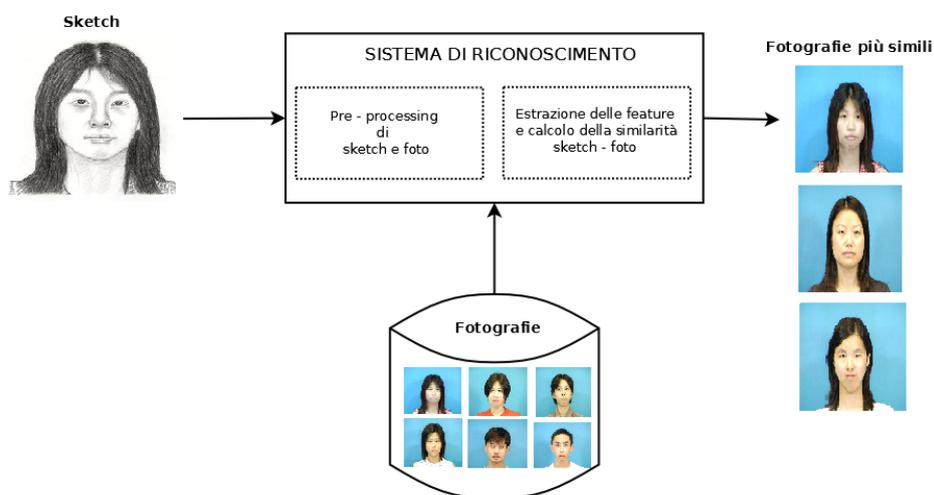


Figura 3.1: Schema a blocchi del sistema di riconoscimento

immagini che contengono solamente i contorni dei volti. Nel concreto, è proprio l'estrazione preliminare dei contorni su entrambe le tipologie di immagini che consente di considerare sullo stesso piano concettuale sia le foto che gli sketch e operare il conseguente calcolo delle feature.

Le feature di forma non sono però le sole ad essere impiegate, bensì ad esse viene abbinato l'utilizzo di un altro particolare descrittore chiamato SURF (*Speeded - Up Robust Features*). Come vedremo dettagliatamente nel Capitolo 5, a differenza delle feature di forma, la tecnica SURF viene attuata direttamente sulle immagini originali ed è quindi in grado di catturare importanti informazioni di tessitura.

Nelle sezioni che completano questo capitolo vedremo a scopo introduttivo sia la fase di pre - processing, che comprende la fase fondamentale di estrazione dei contorni, sia l'elencazione delle varie tipologie di descrittori utilizzati.

## 3.2 Pre - processing del volto

Come anticipato nel precedente paragrafo, per poter effettuare il riconoscimento del volto è necessario innanzi tutto rendere lo sketch e la fotografia come elementi che possano essere direttamente confrontati tra loro. La fase di pre - processing, che viene svolta *una tantum*, consente di raggiungere questo obiettivo assumendo quindi un ruolo di fondamentale importanza, visto che il risultato di questa elaborazione influenzerà in modo determinante il riconoscimento stesso. Senza entrare in dettagli tecnici che verranno esaminati nel Capitolo 4 ma per comprendere comunque ad alto livello in cosa consiste la fase di pre - processing, può essere utile prendere in esame lo schema a blocchi riportato in Figura 3.2. In questa Figura l'input è costituito da uno

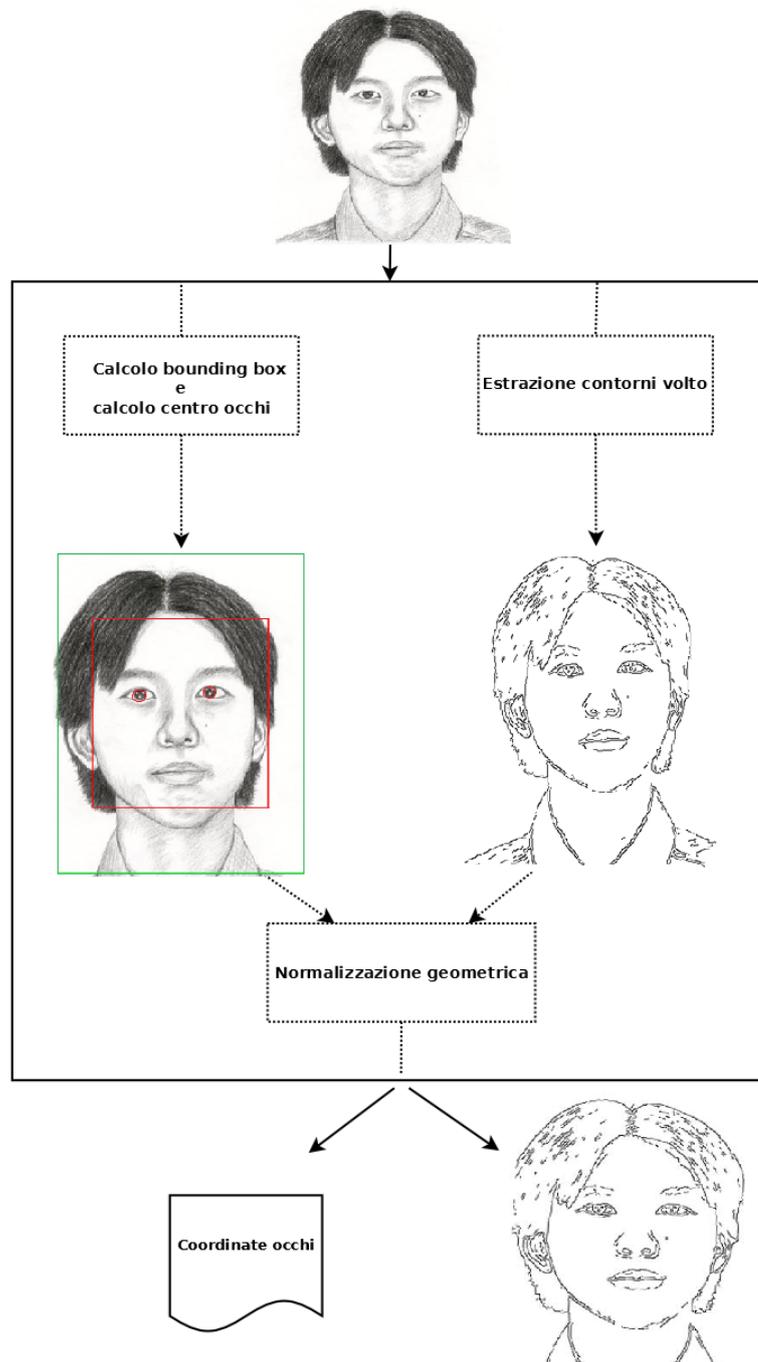


Figura 3.2: Schema a blocchi riassuntivo della fase di pre - processing con input costituito da uno sketch del volto

sketch del volto, ma i passaggi sono i medesimi qualora l'input sia costituito da una fotografia reale. In particolare, le attività intermedie realizzate sono le seguenti:

- determinazione delle coordinate e delle dimensioni del cosiddetto *bounding box*, cioè del rettangolo (colorato in verde in Figura 3.2) che contiene al proprio interno l'intera figura del volto. In questa fase si calcolano anche le coordinate dei centri degli occhi;
- estrazione dei contorni (*edge*) del volto a partire dall'immagine originale;
- normalizzazione geometrica del volto consistente in un allineamento e in un ridimensionamento.

L'output del complessivo pre - processing consiste quindi in due elementi fondamentali: da un lato l'immagine relativa al volto di input normalizzata contenente i soli contorni, dall'altro le informazioni cartesiane relative alla posizione del centro degli occhi in questa nuova immagine. Il calcolo dei contorni dell'immagine, come già anticipato, è un'attività chiave per il riconoscimento visto che consente di aggirare l'ostacolo rappresentato dalla differente modalità di rappresentazione di uno sketch rispetto alla foto. A titolo d'esempio, in Figura 3.3 possiamo notare uno sketch ed una foto afferenti ad uno stesso soggetto, mentre in Figura 3.4 è riportata una coppia sketch - foto relativa a soggetti diversi. In entrambi i casi il lettore può notare come dalle immagini contenenti i soli contorni si possano ancora percepire somiglianze e/o differenze dello sketch rispetto alla foto e viceversa, pur avendo queste immagini un contenuto informativo minore rispetto alle immagini originali. Il cuore dello studio effettuato consiste proprio nell'individuare metodologie e tecniche automatiche in grado di rilevare e quantificare differenze e somiglianze tra volti di cui si conoscono solo i contorni.

Un altro aspetto importante che occorre menzionare è l'utilità dell'attività di normalizzazione geometrica. Essa viene effettuata con lo scopo di produrre immagini in cui i volti abbiano dimensioni standard e siano tutti allineati alla stessa maniera in modo da facilitare il confronto in fase di riconoscimento. Per effettuare questo tipo di elaborazione risulta basilare la rilevazione dei centri dei due occhi nell'immagine originale, le cui coordinate saranno poi opportunamente mappate nella nuova immagine normalizzata e restituite in output. Come vedremo più avanti, infatti, per il calcolo delle feature sarà importante disporre di alcuni punti di riferimento all'interno del volto e queste informazioni cartesiane saranno utili per lo scopo.



Figura 3.3: Comparazione del risultato di estrazione dei contorni a partire da uno sketch e da una foto afferenti al medesimo soggetto



Figura 3.4: Comparazione del risultato di estrazione dei contorni a partire da uno sketch e da una foto afferenti a soggetti diversi

### 3.3 Riconoscimento mediante estrazione di feature

#### 3.3.1 Feature di forma

Dopo aver mostrato a titolo introduttivo l'attività di pre - processing è possibile dedicare la nostra attenzione al cuore del sistema di riconoscimento studiato e implementato, consistente nell'estrazione di particolari feature. In questo paragrafo concentreremo in particolare la nostra attenzione sulle cosiddette feature di forma mentre nel paragrafo successivo verrà mostrata la già citata feature SURF; le varie feature di forma da un lato e SURF dall'altro forniscono assieme un importante contributo per il riconoscimento, in quanto i due tipi di feature sono molto diversi e permettono quindi di estrarre informazioni differenti dall'immagine.

Le feature di forma, probabilmente meglio conosciute nella loro accezione inglese come *shape descriptors*, possono essere viste come un insieme di valori numerici che vengono calcolati per descrivere in modo concreto e oggettivo le caratteristiche salienti che contraddistinguono le figure, generalmente irregolari, contenute all'interno di un'immagine. Tali feature trovano larga applicazione nei sistemi *Content - Based Image Retrieval* (CBIR), cioè quei sistemi che tentano di individuare all'interno di un ampio database di immagini quella/e più somigliante/i ad una certa immagine fornita in input, chiamata *query image*. Questi sistemi basano la propria indicizzazione e ricerca sull'analisi del contenuto delle immagini stesse, contenuto che può essere di diverso tipo; non solo la forma come nel caso di nostro interesse, ma anche il colore, la tessitura o qualsiasi altra informazione che possa essere derivata dall'immagine stessa. Nel corso degli anni studiosi e ricercatori hanno profuso molte energie nello studio dei sistemi CBIR sempre partendo da un presupposto: utilizzare come query image un'immagine dello stesso tipo di quelle contenute nel database. Recentemente, invece, è stata sempre più presa in esame la possibilità di considerare come immagine query non una fotografia reale dell'oggetto ma un suo disegno, dando vita ai cosiddetti sistemi *Sketch - Based Image Retrieval* (SBIR) ed esaltando l'utilità delle feature di forma stesse. Sotto questo punto di vista è interessante notare il contributo dei ricercatori italiani A. Del Bimbo e P. Pala [24] oltre che lo studio dei ricercatori giapponesi T. Hisamori e G. Ohashi [25] i quali hanno progettato un sistema di reperimento di immagini da database a partire da un disegno tracciato dall'utente. Molto interessante anche lo studio di K. Hildebrand *et al.* [26] riguardante le tecniche di valutazione delle performance dei sistemi SBIR e il particolare descrittore studiato da M. Eitz *et al.* [27] in grado di confrontare l'immagine binaria dell'oggetto sotto forma di sketch e le fotografie reali a disposizione nel database.

La panoramica appena effettuata potrebbe indurre il lettore a chiedersi che attinenza abbia tutto ciò con il riconoscimento del volto; in effetti, questo argomento non è trattato in nessuno dei documenti appena citati e i sistemi di

image retrieval basati su sketch che sono stati menzionati (e nei quali si colloca l'utilizzo delle feature di forma) sono studiati per funzionare su database di immagini che poco o nulla hanno a che vedere con la tematica affrontata in questa tesi. In realtà, il punto è proprio questo: le feature di forma sono state e sono tutt'ora utilizzate per il riconoscimento di oggetti generici e per discriminare l'immagine che contiene un oggetto rispetto a quella che ne contiene un altro, mentre è nostro interesse prendere 'in prestito' questa idea e provare ad esplorare l'effettiva utilità delle feature di forma per il riconoscimento del volto da sketch. L'anello di congiunzione tra il concetto di feature di forma e la tematica del riconoscimento da sketch è rappresentato dal fatto che ogni singola feature di forma verrà calcolata a partire dalle immagini contenenti i soli contorni dei volti, cioè dal risultato della fase di pre - processing. L'intuizione è quella di considerare i contorni dei volti come delle forme geometriche, provando a utilizzare specifiche feature per discriminare una faccia rispetto all'altra. L'ambito di lavoro cambia pertanto in modo piuttosto netto rispetto ai sopra citati sistemi di image retrieval CBIR e SBIR: un conto è individuare delle feature che consentano di distinguere una forma dall'altra come succede nei sistemi di image retrieval mentre tutt'altra questione è capire quali feature possano essere adeguate al riconoscimento di volti che basilarmente hanno tutti pressapoco la stessa forma e si differenziano soltanto grazie a determinati particolari locali o globali. Inoltre, è altresì importante sottolineare come molte feature di forma siano pensate per funzionare su contorni chiusi e regolari mentre il volto è caratterizzato da contorni irregolari e spezzettati.

In letteratura esistono numerosissime feature di forma, molte delle quali sono raccolte in un interessante documento redatto da M. Yang *et al.* [28] che è stato preso come riferimento per la scelta di gran parte delle feature di forma utilizzate. Nel contesto di questa tesi ne sono state considerate in tutto 8 e sono di seguito elencate in rigoroso ordine alfabetico:

- Beam Angle Statistics;
- Chain Code;
- Fourier Descriptors;
- Gray Level Profile;
- Local Binary Pattern;
- Local Orientation Histogram;
- Pixel Decimal Value;
- Shape Matrix.

Ognuna di queste feature di forma, il cui significato verrà esaminato in maniera dettagliata nel Capitolo 5, permette di dare una descrizione numerica

di una particolare caratteristica del volto a partire dai suoi contorni. Occorre sottolineare come ogni feature di forma debba avere associato un criterio per quantificare la similarità tra il descrittore calcolato sui contorni di uno sketch e il medesimo descrittore calcolato sui contorni estratti da una fotografia, come riassume lo schema a blocchi riportato in Figura 3.5. Come mostra inoltre Figura 3.7, confrontando lo sketch in input con ogni singola foto presente nel database e ripetendo questa operazione per ogni singola feature di forma, con annesso il contributo della feature SURF introdotta nel successivo paragrafo, sarà possibile ottenere la desiderata classifica delle fotografie più somiglianti allo sketch preso come riferimento. Naturalmente, anche questi dettagli verranno studiati nel Capitolo 5.

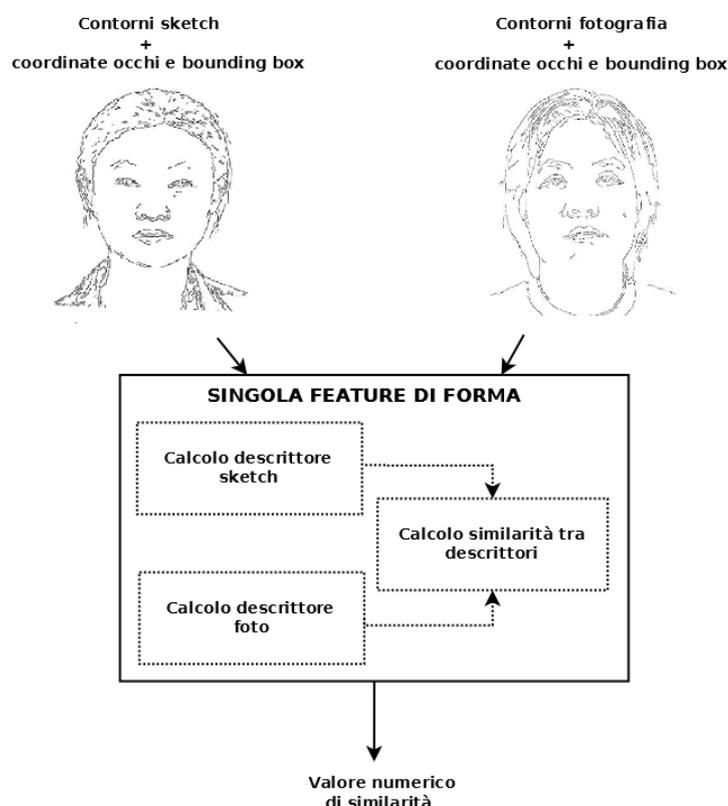


Figura 3.5: Schema a blocchi riassuntivo del calcolo di una feature di forma

### 3.3.2 SURF (Speeded - Up Robust Features)

Il sistema di riconoscimento che è stato studiato e implementato in questa tesi è in gran parte imperniato sul calcolo delle feature di forma a partire dalle immagini contenenti contorni di sketch e foto. In realtà, come anticipato nel Paragrafo 3.1, le feature di forma non sono le sole ad essere state utilizzate;

il loro contributo va ad aggiungersi a quello di una ulteriore feature chiamata SURF [29]. Questa è completamente diversa dalle feature di forma precedentemente introdotte e la prima importante differenza è evidenziata nello schema a blocchi in Figura 3.6. Come si nota, a differenza delle feature di forma, la feature SURF per poter restituire un valore di similarità richiede che in input vengano sottoposte direttamente le immagini originali di sketch e foto normalizzate e non le immagini dei contorni.

La feature SURF, il cui acronimo sta per Speeded - Up Robust Features, rappresenta una versione più veloce e migliorata dell'algoritmo SIFT *Scale Invariant Feature Transform* proposto da D. Lowe nel 1999 [30]. SURF è studiata per permettere l'individuazione e la descrizione di regioni locali di un'immagine e le sue applicazioni sono molteplici includendo il riconoscimento di oggetti, la mappatura per la navigazione robotica, la videosorveglianza, il matching di oggetti in movimento etc. L'intuizione su cui si fonda è quella di calcolare, per ogni oggetto presente in una certa immagine, alcuni punti 'interessanti', detti *keypoint*, in grado di fornire una descrizione caratteristica dell'oggetto. Perché questo procedimento sia affidabile è importante che le caratteristiche estratte dall'immagine siano invarianti ai cambiamenti di scala, di rotazione e in parte invarianti anche ai cambiamenti di illuminazione. SURF permette di identificare in modo corretto gli oggetti grazie ad un algoritmo strutturato in due fasi e riassunto sempre nello schema di Figura 3.6: una prima elaborazione nella quale si individuano i punti d'interesse ed una seconda nella quale si tenta di 'accoppiare' i punti individuati nell'una e nell'altra immagine al fine di scoprire eventuali affinità.

Come successo anche per le feature di forma, siamo di fronte ad una particolare tipologia di feature utilizzata in svariati campi ma ancora poco utilizzata nell'ambito del riconoscimento del volto; il motivo per cui è stata presa in esame anche questa feature è quello di valutare l'effettiva capacità dell'algoritmo di confrontare direttamente immagini che contengono volti in differenti modalità di rappresentazione, come appunto una foto e uno sketch.

I dettagli di questa feature e il modo in cui vengono effettuati gli accoppiamenti tra punti verranno trattati nel Capitolo 5. Ciò che occorre evidenziare a conclusione di questo capitolo è lo schema riportato in Figura 3.7; esso mostra come le feature di forma e la feature SURF procedano indipendentemente le une rispetto alle altre e come il punto di congiunzione sia rappresentato dalla fusione dei risultati di similarità ottenuti per mezzo di ogni singola feature. Solo in questo modo è possibile produrre come risultato finale una classifica delle fotografie più simili allo sketch selezionato in input.

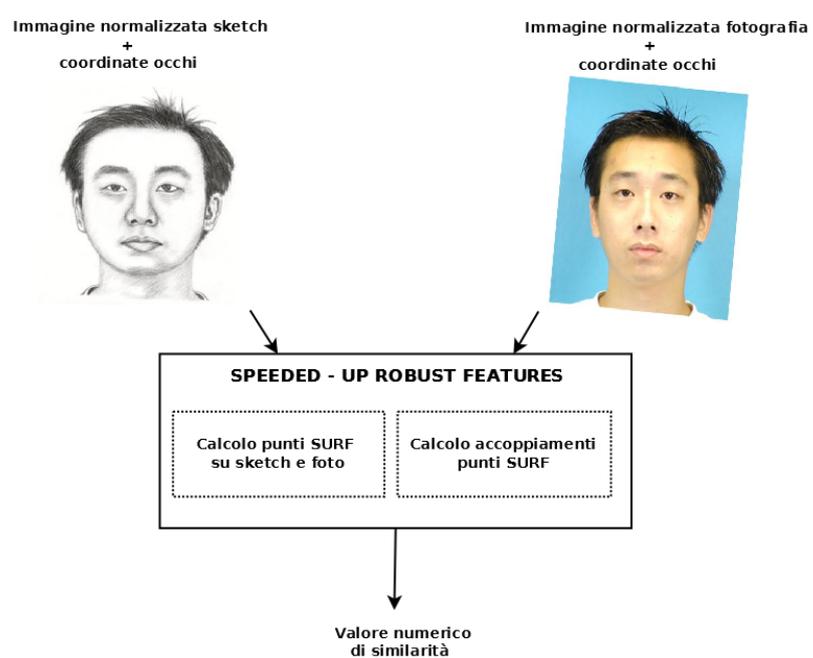


Figura 3.6: Schema a blocchi riassuntivo relativo al calcolo della feature SURF

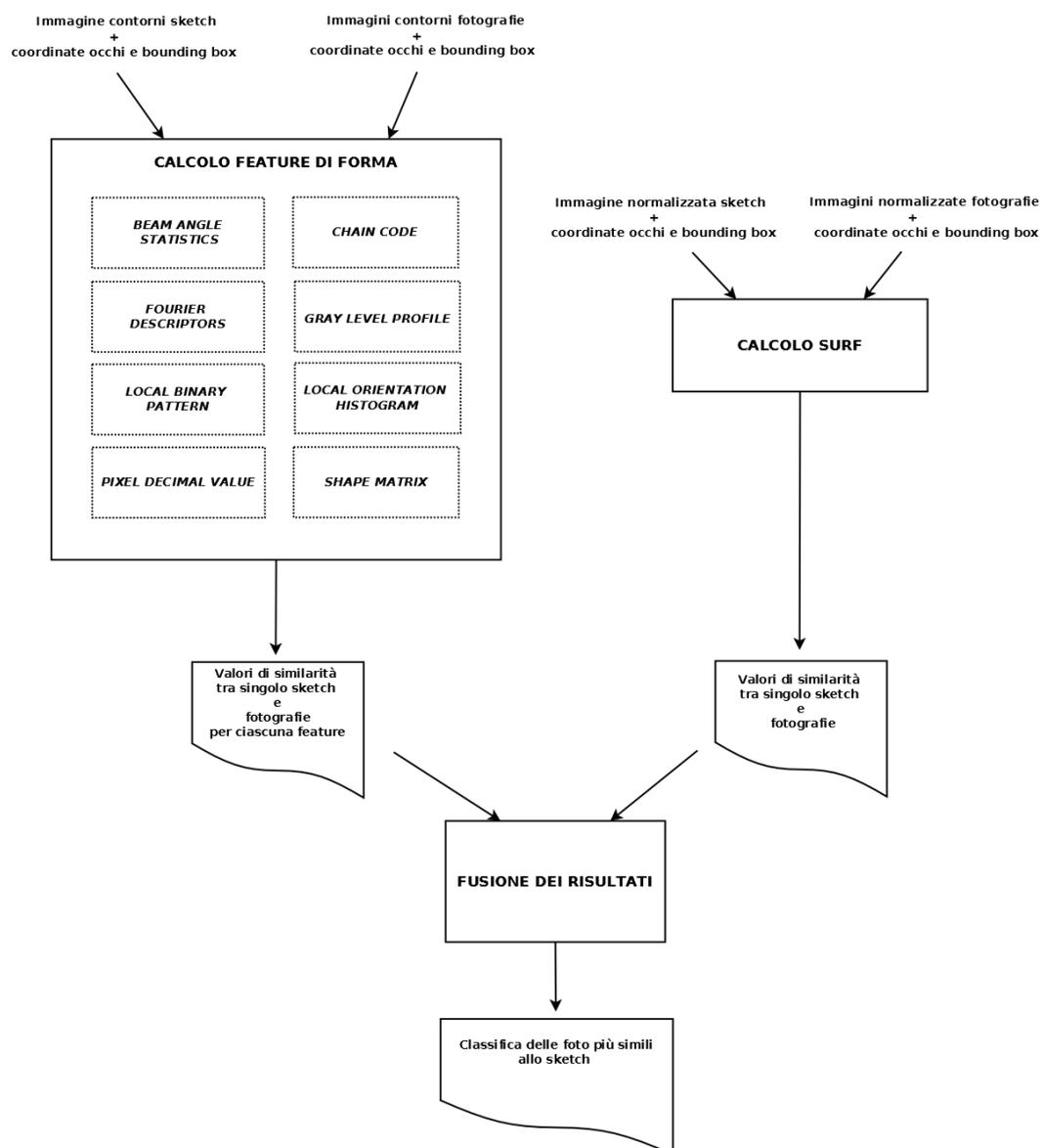


Figura 3.7: Schema a blocchi riassuntivo relativo all'individuazione delle fotografie più somiglianti ad un dato sketch

## Capitolo 4

# Pre - processing del volto

### 4.1 Estrazione dei contorni: IVP Framework

Nel precedente capitolo è stata illustrata ad alto livello la fase di pre - processing evidenziando come essa sia composta dai seguenti blocchi operativi:

- calcolo delle coordinate cartesiane del bounding box e dei centri degli occhi;
- estrazione dei contorni del volto;
- normalizzazione geometrica.

In questo momento il nostro interesse si focalizza sull'attività di estrazione dei contorni, la quale rappresenta uno snodo cruciale per le successive elaborazioni; i contorni estratti consentono infatti di trattare le fotografie e gli sketch come se fossero immagini dello stesso tipo agevolando il calcolo delle già menzionate feature di forma. Come il lettore attento avrà notato, dallo schema a blocchi della già indicata Figura 3.2 si evince che l'estrazione dei contorni viene eseguita a partire dalle immagini originali e i risultati ottenuti sono sottoposti ad una successiva attività di normalizzazione geometrica. Ne consegue che per l'estrazione dei contorni è necessario adottare degli algoritmi che creino meno artefatti possibili, al fine di evitare che questi possano essere accentuati nella seguente fase di normalizzazione. In letteratura esistono numerosi algoritmi di estrazione dei contorni, tra i quali il più famoso e utilizzato risulta essere quello progettato nel 1986 da John Canny [31]; esso utilizza un metodo di calcolo multi - stadio per individuare contorni di molti dei tipi di oggetti normalmente presenti nelle immagini reali. Questo tipo di tecnica, seppur molto impiegata nei contesti più svariati, incorre nel problema prima menzionato legato alla comparsa di artefatti.

In virtù di quanto detto, la scelta della tecnica di estrazione è caduta sull'impiego di un particolare ambiente di elaborazione di immagini chiamato *Image*

*and Vector Processing Framework* (IVPF) [32] [33]. Questo framework, progettato nel 2007 da Jan Klima e Tomas Skopal ed implementato in ambiente .NET 2.0, nasce con un duplice scopo:

- fornire uno strumento versatile ed efficace di estrazione delle forme dalle immagini, in grado di poter essere utilizzato in scenari anche molto diversi tra loro;
- trattare concettualmente come ‘oggetti’ non solo i dati coinvolti in operazioni di image - processing (immagini a colori, a scala di grigio, binarie, mappe di gradienti, vettori, polilinee, ecc.) ma anche gli algoritmi che lavorano su di essi.

In particolare, il framework mette a disposizione una serie di funzionalità già implementate che dal punto di vista dell’utente possono essere considerate come delle *black - box* che ricevono in input e restituiscono in output ben determinate tipologie di dati. Questi algoritmi svolgono delle semplici e ben determinate funzioni ‘base’, che possono poi essere composte assieme al fine di dar vita ad un algoritmo più complesso e strutturato. L’impostazione data dai creatori del framework permette al progettista di creare una nuova procedura di estrazione delle forme trattandola come se fosse un processo funzionale composto da vari sotto processi elementari; esso non deve fare altro che selezionare alcune tra le funzionalità predefinite nel framework, definirne i parametri (quando previsto) e connetterle tra loro dando vita ad una rete di elaborazione in cui i dati si propagano da un blocco funzionale all’altro. L’utente del framework può inoltre fruire di una semplice interfaccia grafica che consente di creare la rete e di verificarne l’effettivo funzionamento, come mostrato ad esempio in Figura 4.1. Nella parte sinistra dell’applicazione sono presenti i nomi dei vari blocchi funzionali utilizzabili e tra questi possiamo notare il filtro Gaussiano, il calcolo del gradiente, la soppressione dei non - massimi e altre tecniche molto utilizzate nelle tipiche applicazioni di estrazione dei contorni. Entrare ulteriormente nei dettagli del framework IVP esulerebbe dal contesto di questa tesi, mentre è sicuramente più attinente ed interessante mostrare le reti che sono state utilizzate per l’estrusione dei contorni dei volti. A tal proposito, è bene specificare che sono state create due reti distinte, una per l’estrusione dei contorni a partire dalle fotografie reali del soggetto ed una per l’estrusione dei contorni a partire dagli sketch. Le due reti, però, sono identiche in quanto a composizione e si differenziano solamente per il valore di alcuni parametri ottimizzati sperimentalmente su un training set separato dal set di immagini usate per testare il sistema. In Figura 4.2 è riportata l’eguale rappresentazione grafica delle due reti composte dai seguenti blocchi funzionali:

- *ImageFromFile* e *LDToFile*: la loro funzione consiste rispettivamente nel caricare un’immagine da file e nel salvare il risultato finale;

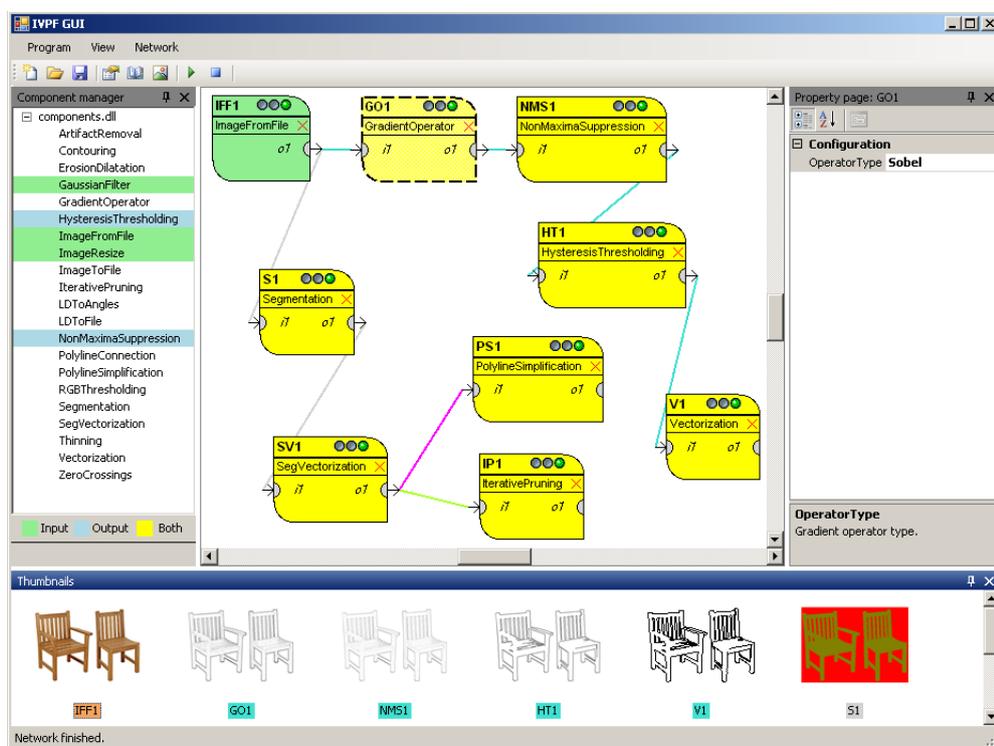


Figura 4.1: Interfaccia grafica del framework IVP e generico esempio di rete funzionale per l'estrazione dei contorni di un'immagine

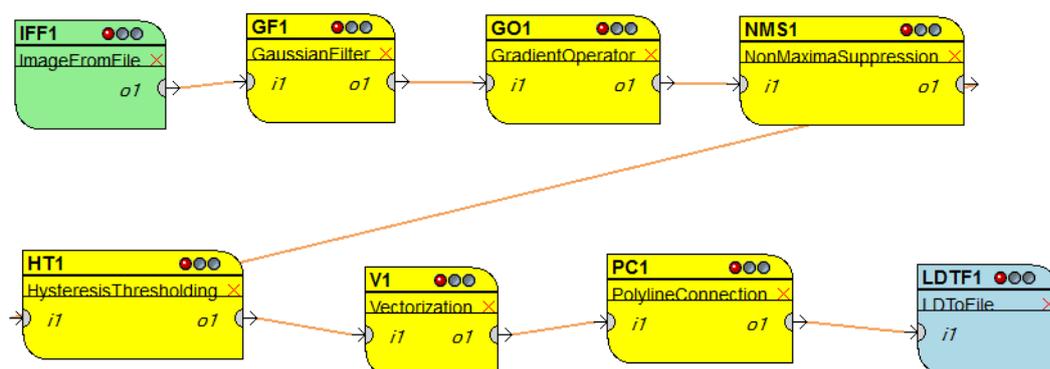


Figura 4.2: Composizione della rete funzionale per l'estrazione dei contorni da foto e sketch

- *GaussianFilter*: nell'elaborazione di immagini si rende spesso necessaria un'operazione di *smoothing* preliminare, volta a eliminare piccole imperfezioni dovute alla presenza di rumore. Il blocco funzionale denominato *GaussianFilter* svolge questo specifico compito, ricevendo in input un'immagine RGB e restituendo un'immagine dello stesso tipo. L'unico compito che spetta al progettista della rete è quello di impostare i valori dei due parametri caratteristici di questa operazione, cioè la dimensione della finestra di convoluzione (*WindowSize*) e la varianza della Gaussiana ( $\sigma$ ). A tal proposito, per la rete di elaborazione dedicata alle fotografie la dimensione della finestra è stata fissata a 11 e  $\sigma$  è stato posto a 2, mentre nella rete relativa agli sketch i due valori sono stati rispettivamente fissati a 3 e 1;
- *GradientOperator*: prima di procedere con l'estrazione dei contorni, è necessario calcolare la cosiddetta mappa del gradiente, cioè il valore del gradiente in ogni punto dell'immagine a partire dai valori di intensità dei singoli pixel. Questo compito viene eseguito dal blocco *GradientOperator* facendo scorrere su tutta l'immagine un filtro derivativo. Il filtro permette di calcolare in ogni punto dell'immagine RGB in input il valore delle derivate parziali lungo l'asse X e lungo l'asse Y che corrispondono alle due componenti del gradiente. La scelta del filtro derivativo spetta al progettista della rete e può ricadere sull'operatore di Roberts, su quello di Prewitt o su quello di Sobel. In particolare, proprio quest'ultimo tipo di filtro è stato scelto per entrambe le reti di elaborazione;
- *NonMaximaSuppression*: il blocco funzionale di soppressione dei non - massimi ha il compito di eliminare, dalla mappa del gradiente prodotta al precedente step, quei pixel che non sono massimi locali rispetto all'orientazione del gradiente stesso. Tale tipo di elaborazione produce una nuova mappa del gradiente in cui però scompaiono eventuali addensamenti con elevati valori del gradiente e questo consente di ottenere degli edge dallo spessore pressochè unitario;
- *HysteresisThresholding*: al fine di selezionare solo gli edge significativi (tralasciando edge 'spuri'), ma evitando allo stesso tempo la frammentazione, si utilizza il concetto di sogliatura con isteresi impiegando due soglie  $T_1$  e  $T_2$ , con  $T_1 > T_2$  per scremare ulteriormente i massimi locali ottenuti nella fase precedente. In particolare, sono inizialmente considerati validi solo i pixel in cui il modulo del gradiente è superiore a  $T_1$ ; i pixel il cui modulo è inferiore a  $T_1$  ma superiore a  $T_2$  sono invece considerati validi solo se adiacenti a pixel validi. Il prodotto di questa elaborazione è ancora una mappa del gradiente. Nella rete di elaborazione relativa alle foto risulta essere  $T_1 = 15$  e  $T_2 = 5$ , mentre nella rete per gli sketch si ha  $T_1 = 30$  e  $T_2 = 20$ ;

- *Vectorization*: questo componente trasforma un'immagine binaria in un insieme di polilinee o poligoni utilizzando il metodo *pixel - chain* proposto in [34] e [35]. Per eseguire questo tipo di operazione si effettua una scansione dell'immagine punto per punto; ogni volta che si incontra un pixel di foreground si decide di connetterlo ai vicini già esaminati sulla base delle regole di connessione adottate. In particolare, esistono due tipologie di connessioni: la connessione di fine linea, quando si effettua la connessione con un pixel finale, e la connessione di giunzione, cioè punti con più di due collegamenti. Effettuate le connessioni si restituisce in output un oggetto che contiene tutte le linee costruite con questo tipo di algoritmo;
- *PolylineConnection*: a causa del rumore introdotto dalle varie elaborazioni, rilevazioni imprecise dei contorni o sovrapposizioni, alcune polilinee significative potrebbero risultare disconnesse in molteplici parti. Si rende quindi necessaria una ulteriore fase di affinamento, chiamata connessione delle polilinee, che effettua una scansione di tutte le linee già disegnate al fine di individuare ulteriori connessioni possibili. Questo blocco funzionale, a causa della sua complessità, prevede il setting di numerosi parametri che descrivono i vincoli in base ai quali un punto debba essere o meno connesso ad un suo vicino (angolo massimo, angolo minimo, validità delle intersezioni, distanza tra polilinee, ecc.); in entrambe le reti questi parametri risultano essere i medesimi.

I risultati intermedi che si ottengono applicando questi diversi blocchi funzionali sono riportati in Figura 4.3 ed in Figura 4.4, rispettivamente per fotografie e sketch. Come si nota, a partire dalle immagini iniziali di sketch e foto, le attività appena descritte producono come risultato immagini contenenti contorni del volto il più possibile continui e connessi tra loro.

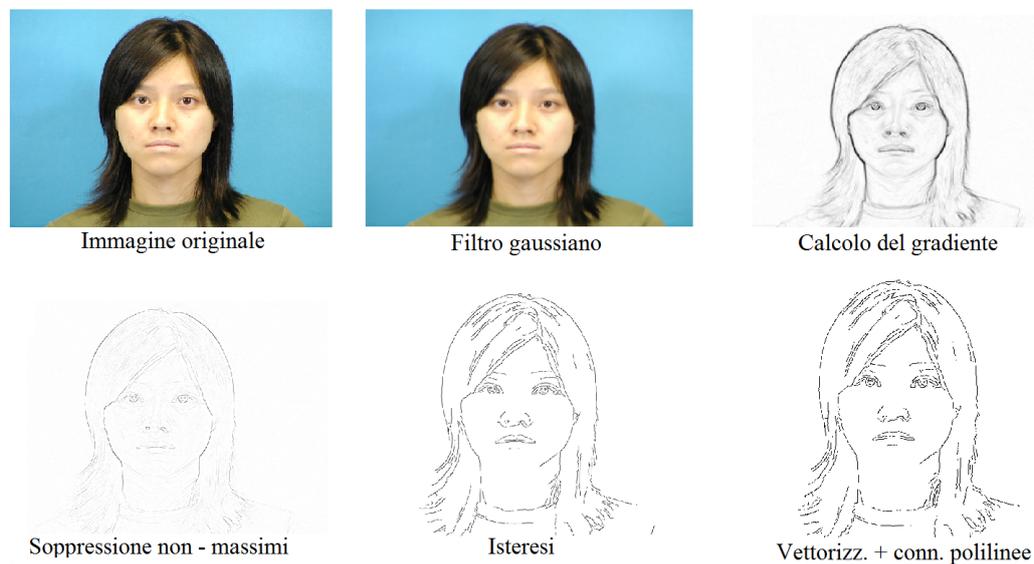


Figura 4.3: Risultati intermedi degli step di estrazione dei contorni a partire da una fotografia

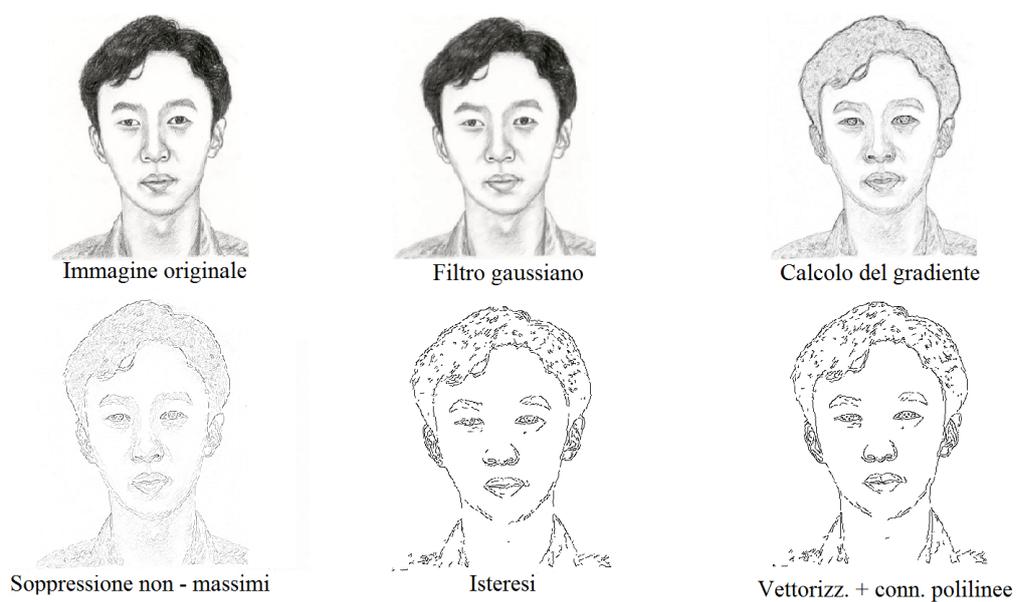


Figura 4.4: Risultati intermedi degli step di estrazione dei contorni a partire da uno sketch

## 4.2 Normalizzazione: BioLab - ICAO Framework

L'attività di estrazione dei contorni precedentemente illustrata restituisce un'immagine che, pur avvicinandosi a ciò che si desidera ottenere, deve essere ulteriormente perfezionata. Questa necessità è legata al fatto che i volti contenuti nei dataset di sketch e foto considerati come casi di studio (vedi Capitolo 7), pur essendo in posizione frontale ed in primo piano, possono essere affetti da piccole inclinazioni dipendenti dalla posa assunta al momento dell'acquisizione come riportato in Figura 4.5.



Figura 4.5: Esempio di un volto con un buon allineamento (a sinistra) e di un volto che necessita di essere allineato (a destra)

Inoltre, le dimensioni dei volti contenuti nelle foto e negli sketch potrebbero avere entità diverse tra loro in base ai parametri della macchina fotografica o alle decisioni dell'artista. Per tutti questi motivi si rende necessaria una fase di normalizzazione che prevede da un lato l'allineamento dei volti e dall'altro un ridimensionamento a valori standard e stabiliti in via preliminare, producendo come risultato una nuova immagine dei contorni normalizzata e quindi omogenea a tutte le altre. Indubbiamente, esistono metodi diversi e varie tecniche per effettuare questa trasformazione; nella fattispecie di questa tesi, l'intuizione che sta alla base della normalizzazione consiste nello sfruttare due riferimenti tipici del volto:

- coordinate cartesiane dei due occhi: rilevando il centro degli occhi è possibile calcolare l'angolo che insiste tra l'asse che congiunge i due centri e l'asse orizzontale. Se questo angolo è nullo il volto è già perfettamente allineato, mentre se esso è diverso da zero si rende necessaria una rotazione del volto volta ad annullare l'angolo e produrre l'allineamento desiderato (Figura 4.6);

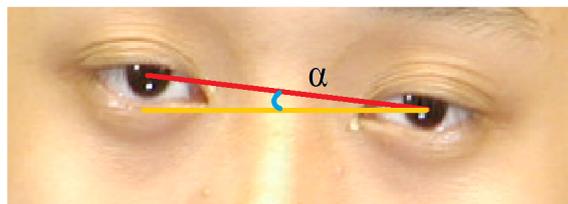


Figura 4.6: Esempio di angolo che sussiste tra l'asse che congiunge i due centri degli occhi e l'asse orizzontale

- bounding box del volto: si intende il rettangolo che include al proprio interno l'intera figura del volto. Calcolando le dimensioni del bounding box si può valutare di quanto queste si discostino da quelle standard e si può conseguentemente procedere ad un ridimensionamento.

Il calcolo dei centri degli occhi e l'individuazione del rettangolo del volto sono due operazioni complesse ma già studiate ed implementate attraverso svariate tecniche. In questa occasione la scelta è caduta sull'impiego di alcune funzionalità già presenti in un framework di lavoro chiamato BioLab - ICAO progettato e curato da M. Ferrara, A. Franco, D. Maio e D. Maltoni [36]. Questo framework nasce con l'obiettivo di fornire strumenti di verifica di conformità alle normative previste dallo standard ISO / ICAO (*International Civil Aviation Organization*) per le fotografie presenti nei passaporti. Esso contiene l'implementazione in ambiente .NET di numerose funzioni per il testing dell'adeguatezza delle caratteristiche del volto presente in una fotografia; tra queste funzioni c'è anche il rilevamento dei centri degli occhi e il calcolo del bounding box. Il motivo principale per cui la scelta è caduta su questo framework, oltre all'opportunità di sfruttare qualcosa di già disponibile e ampiamente collaudato oltre che perfettamente funzionante, è legato alla buona capacità di individuare le informazioni richieste anche sugli sketch. Le funzionalità del framework che sono state sfruttate consentono infatti, grazie alle particolari tecniche utilizzate, di individuare occhi e bounding box non solo nelle comuni immagini fotografiche ma anche nei disegni dei volti.

Per quanto concerne il calcolo del bounding box, esso si ottiene attraverso una fusione pesata dei risultati ottenuti attraverso l'applicazione dell'algoritmo proposto da Maio e Maltoni in [37] e da quello di Viola - Jones descritto in [38]. La combinazione di questi due metodi consente di ottenere un bounding box come quello riportato in rosso in Figura 4.7. Come si può notare, oltre alla capacità del metodo di lavorare sugli sketch, il rettangolo rosso racchiude il volto in maniera troppo stringente tralasciando dei dettagli che possono essere molto utili per il successivo riconoscimento; pertanto, il bounding box finale è aumentato percentualmente rispetto a quello calcolato inizialmente, ottenendo un nuovo rettangolo indicato col colore verde.

Per quanto concerne invece la rilevazione del centro degli occhi si parte dal

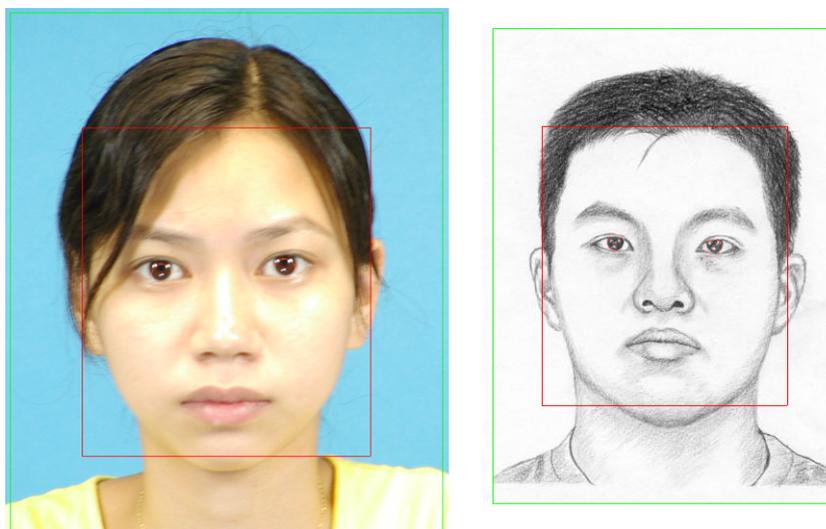


Figura 4.7: Risultato della rilevazione del bounding box del volto

già individuato bounding box. In particolare, si procede all'individuazione dei centri delle due pupille e successivamente al calcolo delle coordinate dei quattro corner degli occhi. Le tecniche sfruttate per queste operazioni sono quelle progettate da Dobes *et al.* in [39] e da Yuille *et al.* in [40], rispettivamente. Il risultato del rilevamento dei centri degli occhi su sketch e fotografie può essere apprezzato in Figura 4.8.



Figura 4.8: Risultato della rilevazione dei centri degli occhi

A questo punto, individuate le informazioni di interesse sulle immagini originali, non resta che procedere al ridimensionamento e all'allineamento dell'immagine dei contorni ottenuta al passo precedente oltre che al mapping delle vecchie coordinate degli occhi sulla nuova immagine normalizzata. Ciò che si

ottiene, come mostra Figura 4.9, è una nuova immagine del tutto simile alla precedente ma trasformata per essere sottoposta alle successive elaborazioni e garantire al tempo stesso la massima invarianza.



Figura 4.9: Immagine contenente i contorni del volto prima della normalizzazione (a sinistra) e dopo la normalizzazione (a destra)

## Capitolo 5

# Calcolo delle feature e riconoscimento

### 5.1 Feature di forma

Conclusa la fase di pre - processing e create le immagini dei contorni di sketch e fotografie opportunamente normalizzate, si può procedere con l'operazione di estrazione delle feature e con la seguente identificazione del soggetto. Questo primo paragrafo, con le sue sottosezioni, è dedicato in particolare all'analisi dettagliata delle feature di forma, che consistono nel calcolo di particolari descrittori a partire dagli edge dei volti. Tali feature, menzionate a titolo introduttivo nel precedente Capitolo 3, nascono in realtà con l'obiettivo di fornire una rappresentazione numerica che possa consentire di discriminare forme diverse tra loro. Non essendo note a priori le performance di queste feature con riferimento al riconoscimento del volto, esse sono state implementate per valutarne l'effettiva bontà: in ognuno dei sottoparagrafi che seguono sarà possibile entrare nel merito di ciascuna feature analizzando l'algoritmo che ne sta alla base e comprendendo quali siano le misure adottate per confrontare un descrittore estratto da uno sketch con uno estratto da una foto. I risultati ottenuti in termini prestazionali tramite la combinazione dei contributi di ciascuna feature sono riportati invece nel Capitolo 7.

Prima di passare all'analisi dettagliata delle feature di forma è bene specificare che pur essendo ognuna distinta dall'altra, ci sono anche degli elementi condivisi. In particolare, un elemento condiviso da gran parte delle feature è il calcolo del bounding box del volto all'interno dell'immagine contenente gli edge. Questo tipo di operazione non va confusa con quanto visto nella fase di pre - processing; in quella occasione il rilevamento del rettangolo includente il volto veniva effettuato sulla fotografia o sullo sketch ed era necessario per estrarre la porzione del volto da sottoporre alle successive elaborazioni di normalizzazione. In questo caso, invece, un nuovo calcolo del bounding si rende necessario per delimitare l'area dell'immagine nella quale sarà necessario operare per il calcolo

della feature. Tale tipo di operazione preliminare è utile anche per ‘scartare’ a priori zone dell’immagine non significative, come ad esempio lo spazio bianco inutilmente presente intorno alla figura del volto, permettendo di risparmiare tempo calcolo nelle successive elaborazioni.

La determinazione del bounding box è un’operazione piuttosto semplice che viene effettuata con una scansione dell’immagine riga per riga alla ricerca dei punti di contorno del volto più lontani possibili dai due centri degli occhi; un possibile esempio di bounding box calcolato è presente in Figura 5.1.

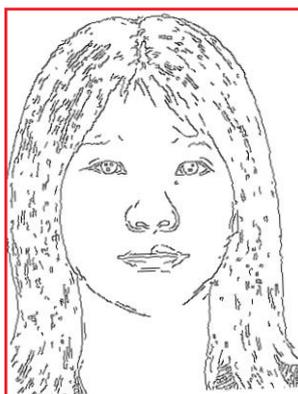


Figura 5.1: Bounding box calcolato a partire dall’immagine dei contorni del volto: l’area inclusa all’interno sarà considerata per l’estrazione delle feature

### 5.1.1 Gray Level Profile

La prima feature di forma ad essere presa in esame è chiamata Gray Level Profile e come si intuisce dal nome il suo scopo è quello di fornire una rappresentazione dei livelli di grigio che contraddistinguono il profilo del soggetto. A tal proposito occorre tenere ben presente che l’immagine su cui lavoriamo, essendo una rappresentazione dei contorni del volto, può essere considerata a tutti gli effetti come immagine a scala di grigio; ogni pixel dell’immagine assume una tonalità che va da 0 (pixel nero) a 255 (pixel bianco). Inoltre, ogni punto dell’immagine può essere classificato in modo esclusivo: se esso assume una colorazione diversa dal bianco (valore pari a 255) esso è un punto facente parte del contorno e quindi denominato punto di *foreground*, mentre è di sfondo (*background*) in caso contrario. Questo aspetto può essere generalizzato introducendo il concetto di soglia di binarizzazione: i valori di intensità superiori a questa soglia sono da considerarsi come appartenenti al background mentre quelli inferiori sono da considerarsi facenti parte del foreground.

L’intuizione su cui si basa la feature chiamata Gray Level Profile consiste nell’utilizzare dei descrittori che possano riassumere in forma compatta l’informazione relativa ai pixel di foreground che costituiscono il volto. In particolare, la

tipologia di descrittore che si è deciso di adottare in questo caso è l'istogramma: questo grafico è molto utilizzato quando si rende necessario registrare il numero di eventi che ricadono in una certa categoria, come ad esempio il numero di persone che abitano nelle città dell'Emilia Romagna, il numero di millimetri di pioggia caduta in ogni mese dell'anno, il numero di pixel di un'immagine contraddistinti da un certo livello di colore e così via. Ogni istogramma è inoltre costituito da diversi rettangoli o *bin*: ogni bin tiene traccia della misura dell'evento in un singolo caso (ad esempio, il numero di pixel di un'immagine che assume la sola colorazione azzurra) o può comunque riferirsi ad un insieme di valori se questi vengono discretizzati. Nel nostro caso l'evento da registrare è rappresentato dal numero di punti di foreground che si trovano lungo una singola linea orizzontale che 'taglia' il volto da sinistra verso destra. Il numero di linee orizzontali su cui quantificare la presenza di pixel di foreground e, conseguentemente, il numero di bin costituenti l'istogramma è un parametro della feature che deve essere stabilito a priori. Infine, per sfruttare meglio questa feature, il volto viene suddiviso esattamente in due metà; ciò comporta che gli istogrammi costruiti per ciascun volto saranno due, uno relativo alla porzione sinistra del volto ed uno relativo alla porzione destra.

Addentrando maggiormente nel dettaglio, l'algoritmo di costruzione dei due istogrammi prevede diversi step, di cui il primo risulta essere proprio il calcolo del valore di ascissa che contraddistingue la metà esatta del volto. Avendo a disposizione le coordinate dei centri degli occhi, la linea verticale che separa in due metà il volto (linea blu verticale in Figura 5.2) è semplicissima da calcolare e risulta essere quella passante per il punto medio tra i due occhi. In altri termini, il valore di ascissa *HalfX* che identifica la metà del volto è ottenuto come:

$$HalfX = \frac{(EyeSx_X + EyeDx_X)}{2}$$

dove *EyeSx<sub>X</sub>* è il valore di ascissa dell'occhio sinistro mentre *EyeDx<sub>X</sub>* il valore di ascissa dell'occhio destro (il valore di ordinata per i due occhi è il medesimo visto che i volti sono allineati).

Effettuata questa operazione preliminare si può passare al calcolo vero e proprio dei bin dell'istogramma. Come detto in precedenza, il valore di ogni bin viene determinato conteggiando il numero di pixel di foreground che insistono su ognuna delle linee orizzontali rappresentate in Figura 5.2. Ogni linea è da considerarsi un 'campione' e il numero di campioni prelevati dall'immagine è stabilito tramite un apposito parametro. Terminata questa attività il risultato prodotto è una coppia di istogrammi, uno relativo alla metà di destra del volto ed uno relativo alla metà di sinistra.

L'ultimo passo consiste a questo punto nel raggruppamento e nella normalizzazione dei bin. Per ottenere degli istogrammi che siano il più possibile rappresentativi delle due metà del volto si rende infatti spesso necessaria la scelta di un fattore di campionamento piuttosto elevato e questo comporta la produzione di istogrammi con un numero altrettanto elevato di bin. Inoltre,

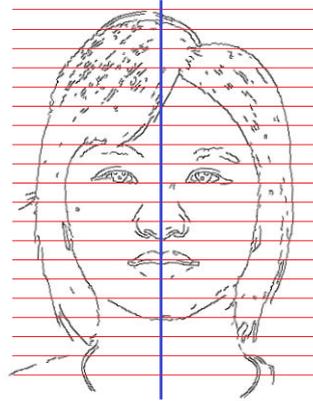


Figura 5.2: Gray Level Profile: la linea blu rappresenta la divisione in due metà del volto, mentre su ognuna delle linee rosse orizzontali si effettua il conteggio dei pixel di foreground

istogrammi estratti da volti distinti risultano differire in valore massimo e valore minimo e questa caratteristica, unita alla eccessiva numerosità dei bin, risulta essere controproducente per il confronto. Occorre pertanto produrre nuovi istogrammi che abbiano un numero inferiore di bin e i cui valori siano opportunamente normalizzati. Tale obiettivo viene ottenuto mediante un primo sottocampionamento consistente nel considerare bin vicini a gruppi di  $n$  e sommandone i valori; se  $m$  sono i bin dell'istogramma originale, il nuovo istogramma presenta  $m/n$  bin. Esso viene poi normalizzato sulla base del proprio valore massimo. In Figura 5.3 possiamo notare un esempio di istogramma originale e del conseguente istogramma normalizzato.

L'algoritmo illustrato riguarda la costruzione degli istogrammi, ma è importante analizzare anche l'aspetto riguardante il confronto tra istogrammi relativi a volti distinti al fine di valutarne la distanza, o al contrario, la similarità. Consideriamo ad esempio in Figura 5.4 e Figura 5.5 due casi di calcolo della feature Gray Level Profile quando il soggetto è lo stesso e quando invece sono trattati due soggetti distinti (rispettivamente): si può notare che in effetti quando il soggetto è il medesimo c'è una certa affinità a livello di andamento degli istogrammi, mentre quando si tratta di soggetti distinti gli istogrammi possono essere anche piuttosto diversi. Definendo con  $HistDx_S$  l'istogramma della porzione destra di uno sketch, con  $HistSx_S$  l'istogramma della parte sinistra di uno sketch, con  $HistDx_F$  l'istogramma della porzione destra di una foto e con  $HistSx_F$  l'istogramma della porzione sinistra di una foto, la misura di distanza  $D$  tra uno sketch ed un fotografia per mezzo dei loro istogrammi è ottenuta come:

$$D = \text{Min}[\text{dist}(HistDx_S, HistDx_F), \text{dist}(HistSx_S, HistSx_F)]$$

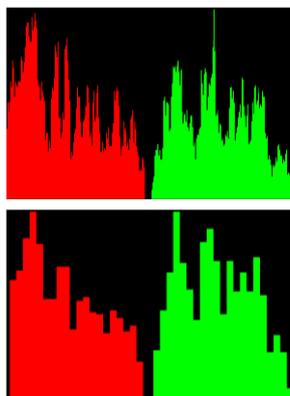


Figura 5.3: Gray Level Profile: istogramma risultante dal processo di conteggio dei pixel di foreground (sopra) e corrispondente istogramma normalizzato a bin raggruppati (sotto)

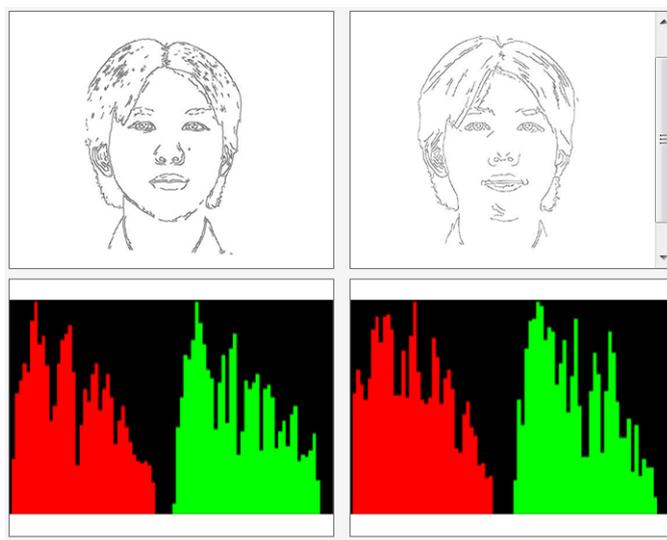


Figura 5.4: Gray Level Profile: istogrammi (in rosso quelli relativi alle parti sinistre dei due volti, in verde quelli relativi alle parti destre) calcolati da uno sketch e da una foto afferenti al medesimo soggetto

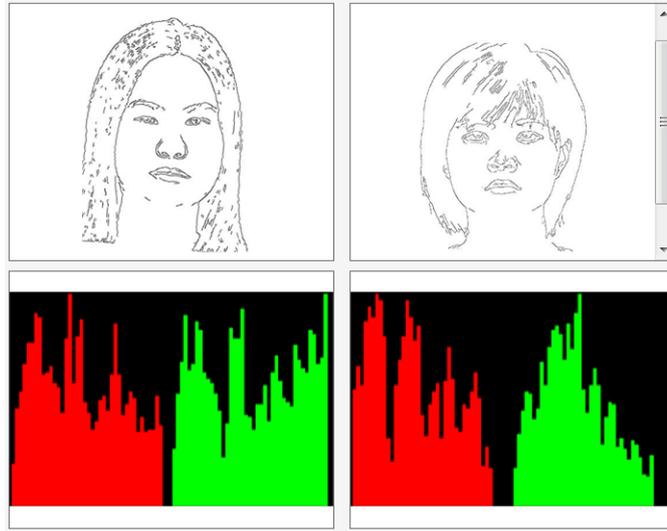


Figura 5.5: Gray Level Profile: istogrammi (in rosso quelli relativi alle parti sinistre dei due volti, in verde quelli relativi alle parti destre) calcolati da uno sketch e da una foto afferenti a soggetti distinti

Resta a questo punto da stabilire quale criterio utilizzare per il calcolo della succitata funzione *dist* la quale quantifica numericamente la diversità di due singoli istogrammi. Prendendo spunto da [41] sono state considerate 4 diverse misure di distanza: la distanza euclidea al quadrato, la divergenza Kullback - Leibler, la distanza chi - quadro e l'intersezione. La distanza euclidea al quadrato  $\varepsilon$  è una misura molto utilizzata in svariati contesti e anche per gli istogrammi è possibile calcolarne il valore; in particolare, a partire da due istogrammi  $H'$  e  $H''$  ed indicando con  $H'(i)$  il valore del  $i$ -esimo bin dell'istogramma  $H'$  (ed analogamente per  $H''(i)$ ), la distanza si calcola applicando la seguente formula:

$$\varepsilon(H', H'') = \sum_{i=1}^d (H'(i) - H''(i))^2$$

Considerando la stessa notazione espressa sopra, è possibile definire la cosiddetta misura di distanza chi - quadro  $\chi_2^2$  e la misura di divergenza KL:

$$\chi_2^2(H', H'') = \sum_{i=1}^d \frac{(H'(i) - H''(i))^2}{H'(i) + H''(i)}$$

$$KL(H', H'') = \sum_{i=1}^d (H'(i) - H''(i)) \ln \frac{H'(i)}{H''(i)}$$

L'ultima misura considerata deriva dalla classica misura di intersezione tra due istogrammi; siccome quest'ultima fornisce in realtà un valore di similarità, è necessario calcolarne il reciproco al fine di ottenere il corrispettivo valore di distanza. L'intersezione  $\cap$  tra due istogrammi è definita come:

$$\cap(H', H'') = \sum_{i=1}^d \min(H'(i), H''(i))$$

Le 4 misure di distanza indicate sono state tutte implementate, sia per permettere all'utilizzatore dell'applicazione di scegliere quella preferita sia per valutare le caratteristiche di ognuna.

### 5.1.2 Shape Matrix

La panoramica delle feature di forma prosegue con la descrizione della seconda feature implementata, denominata Shape Matrix. Come le altre, anche questa nasce per rappresentare in modo numerico e compatto la forma di un generico oggetto. Ad esempio, nel documento redatto da M. Yang *et al.* [28] e preso come riferimento per lo studio e la scelta della maggior parte delle feature, è possibile trovare una doppia versione di Shape Matrix: una chiamata *Square Model Shape Matrix* ed una denominata *Polar Model Shape Matrix*. Nel contesto di questa tesi è stata presa in esame la prima versione, la quale consiste sostanzialmente nel costruire una griglia o matrice  $M \times N$  esattamente sovrapposta alla figura del volto, come mostrato a titolo d'esempio in Figura 5.6. Il numero di righe  $M$  ed il numero di colonne  $N$  sono dei parametri dell'applicazione che devono essere stabiliti in via preliminare dall'utilizzatore.

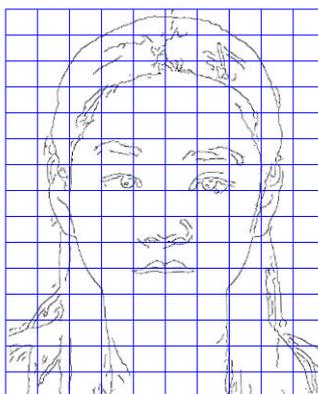


Figura 5.6: Shape Matrix: griglia a 15 righe e 10 colonne sovrapposta all'immagine dei contorni del volto per il calcolo della feature

L'intuizione su cui si basa Shape Matrix è quella di calcolare, per ogni cella di

riga  $j$  e colonna  $k$ , un'informazione numerica  $B_{jk}$  ottenuta a partire dalla porzione di contorno del volto (foreground) inclusa all'interno della cella stessa. Componendo assieme le informazioni numeriche ottenute da ciascuna cella, si ottiene un vettore con tante componenti tante quante sono le celle della griglia: questo vettore rappresenta il descrittore finale del volto e può essere confrontato con altri vettori al fine di valutare la similarità tra soggetti.

Nella parte finale di questo paragrafo verranno esaminate le tecniche di confronto tra vettori, mentre in questo momento l'interesse si concentra sul tipo di informazione che viene ottenuta a partire da ciascuna cella. A tal proposito, è bene specificare che il contesto è identico a quello analizzato per la feature Gray Level Profile: l'immagine dei contorni a nostra disposizione è di tipo grayscale ed ogni pixel può essere considerato come pixel di foreground solo se il livello di grigio ad esso corrispondente è diverso da 255 (bianco), mentre in caso contrario il pixel è da considerarsi come sfondo o background. Ribadire questa distinzione è importante in quanto l'informazione  $B_{jk}$  ricavata da ciascuna cella è pari alla cosiddetta percentuale di *coverage* (copertura) della cella, cioè la percentuale di pixel di foreground rispetto al numero totale di punti contenuti all'interno della cella stessa. Matematicamente, considerando con  $\#F_{jk}$  il numero di pixel di foreground contenuti all'interno della cella  $jk$  presente nella matrici di dimensioni  $M \times N$ , la percentuale di coverage  $\%Cov_{jk}$  è ottenuta come:

$$\%Cov_{jk} = \frac{\#F_{jk}}{M * N} * 100$$

La copertura di una cella permette di fornire una quantificazione numerica e quindi oggettiva della presenza di contorni del volto in una determinata regione e componendo i valori di copertura di tutte le celle si può ottenere un vettore con  $M * N$  componenti che dia una descrizione complessiva del volto sotto questo punto di vista.

Al fine di valutare al meglio le potenzialità di questa feature sono state introdotte alcune varianti al calcolo mostrato in precedenza. La prima variante consiste nell'utilizzo opzionale di un peso da associare a ciascuna cella della matrice. In particolare, l'idea è quella di attribuire un peso maggiore alle celle collocate in zona centrale e di attribuire un peso inferiore alle celle man mano che ci si allontana dal centro della matrice. Questa pesatura permette di attribuire alle zone centrali del volto, come occhi, naso e bocca una importanza maggiore in fase di riconoscimento rispetto ad altre zone periferiche come i capelli, il collo e le orecchie. I pesi associati alle celle variano tra 0 e 1, sono inversamente proporzionali alla distanza dal centro e vanno a moltiplicare la percentuale di coverage. In Figura 5.7 può essere apprezzato visivamente la distribuzione dei pesi all'interno della matrice: il peso applicato alla cella centrale è più elevato dei pesi applicati alle celle periferiche (a colore più scuro corrisponde peso più basso). Un'altra importante variante al calcolo di Shape

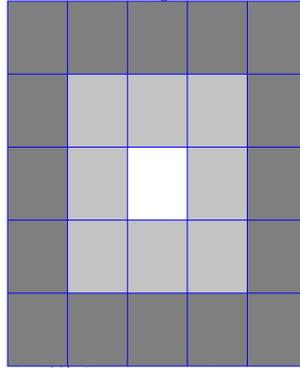


Figura 5.7: Shape Matrix: rappresentazione grafica dei pesi applicati alle celle della griglia (a colore più scuro corrisponde peso più basso)

Matrix prende spunto dal già citato Square Model Shape Matrix descritto in [28] e consiste nell'introduzione di un valore di soglia  $Th$ . L'informazione  $B_{jk}$  associata ad ogni generica cella cambia di conseguenza in questo modo:

$$B_{jk} = \begin{cases} 1, & \text{se } \%Cov_{jk} \geq Th \\ 0, & \text{altrimenti} \end{cases}$$

Questo significa che il vettore finale risultante sarà di tipo binario; in corrispondenza delle celle in cui la percentuale di copertura supera la soglia prevista il valore sarà 1, in caso opposto 0.

Quanto illustrato fino a questo momento fa capire come l'utilizzo della feature Shape Matrix possa potenzialmente produrre in output due tipologie di vettori descrittivi del volto. Nel caso in cui si decida di considerare la sola percentuale di copertura con l'applicazione opzionale della pesatura, il vettore risultante sarà costituito da valori decimali mentre in caso di utilizzo della soglia il valore di ciascuna componente del vettore sarà 0 oppure 1. Questa doppia possibilità di costruzione comporta anche l'utilizzo di criteri diversi per il confronto tra vettori. In particolare, se i valori ottenuti dal calcolo della feature sono decimali è possibile utilizzare la classica distanza euclidea tra vettori. Definendo con  $S = (s_1, s_2, s_3, \dots, s_n)$  il vettore  $n$ -dimensionale ottenuto dall'analisi dei contorni dello sketch e con  $F = (f_1, f_2, f_3, \dots, f_n)$  il vettore  $n$ -dimensionale ottenuto dall'analisi dei contorni della foto, la distanza euclidea  $E$  è calcolata come:

$$E = \sqrt{\sum_{k=1}^n (s_k - f_k)^2}$$

Se invece la scelta dell'utilizzatore dell'applicazione cade sull'adozione di vettori binari e quindi sull'utilizzo della soglia, la misura di distanza tra vettori può

essere effettuata utilizzando la cosiddetta formula di Hamming. La distanza di Hamming prende il suo nome da Richard Hamming, che la introdusse nel suo lavoro sui codici per il riconoscimento e la correzione degli errori nel 1950. Nella teoria dell'informazione, la distanza di Hamming tra due stringhe di uguale lunghezza è il numero di posizioni nelle quali i simboli corrispondenti sono diversi; ad esempio, la distanza tra 1011101 e 1001001 è pari 2. Questo stesso concetto può essere applicato ai vettori binari ottenuti dal calcolo della feature Shape Matrix, quantificando attraverso un calcolo molto efficiente la diversità tra l'immagine dei contorni ottenuta a partire da uno sketch e l'immagine dei contorni ottenuta a partire da una foto.

### 5.1.3 Local Orientation Histogram

La terza feature di forma che andiamo ad esaminare prende il nome di Local Orientation Histogram; essa sfrutta da un lato concetti già visti nei precedenti due paragrafi come l'utilizzo degli istogrammi e la suddivisione dell'immagine dei contorni in celle e dall'altro lato prevede l'introduzione di un nuovo elemento: il calcolo dell'immagine direzionale. La procedura di estrazione di questa feature è piuttosto elaborata ed è pertanto consigliabile procedere a piccoli passi.

Diciamo innanzi tutto che l'utilizzo di questo tipo di feature non è nuovo per il riconoscimento di forme; a titolo d'esempio possiamo citare lo studio effettuato da Saveedra e Bustos [42] nel quale si trova un esempio di applicazione di questa feature nell'ambito dei già citati sistemi SBIR, cioè quei sistemi nei quali l'immagine dell'oggetto da ricercare all'interno del database è disegnata manualmente da un artista o tramite un apposito software. Nel contesto di questa tesi si vuole prendere 'in prestito' questa idea ed applicarla per il confronto dei contorni dei volti al fine di valutarne la somiglianza.

Come già anticipato, la vera novità che contraddistingue questa feature dalle precedenti e che ne rappresenta il cuore di funzionamento è il calcolo della cosiddetta immagine direzionale. L'immagine direzionale è una matrice in cui ogni elemento  $ij$ , corrispondente al nodo  $[i, j]$  di una griglia quadrata posizionata sul pixel  $[x_i, y_j]$ , è un vettore bidimensionale contraddistinto da una direzione e da un modulo. La direzione, chiamata anche orientazione media  $\theta_{ij}$ , è tangente agli edge presenti nell'intorno del pixel. Il modulo, detto anche coerenza (o affidabilità), assume valori elevati in presenza di edge netti e marcati mentre ha valori bassi in caso di presenza di rumore.

L'immagine direzionale viene spesso utilizzata come passo di elaborazione preliminare: ad esempio, Maio e Maltoni in [37] utilizzano il calcolo dell'immagine direzionale per favorire una successiva attività di individuazione di forme ellissoidali approssimanti il volto. Ma non solo: l'immagine direzionale trova larga applicazione nell'ambito del riconoscimento di impronte digitali e in particolare modo per la segmentazione delle impronte, come descritto in [43]. Prima di addentrarci nei dettagli relativi alla produzione dell'immagine direzionale,

possiamo a titolo d'esempio visualizzare in Figura 5.8 il risultato di questo calcolo a partire da due generiche immagini tratte dallo studio effettuato in [37]: come si nota, ogni elemento dell'immagine direzionale è rappresentativo della forma e quindi degli egde presenti in quella regione. Inoltre, il focus sul singolo elemento evidenzia come esso sia contraddistinto dall'angolo  $\theta_{ij}$  e dalla sua ampiezza  $r_{ij}$  chiamata anche *reliability* (affidabilità), proprio come descritto in precedenza.

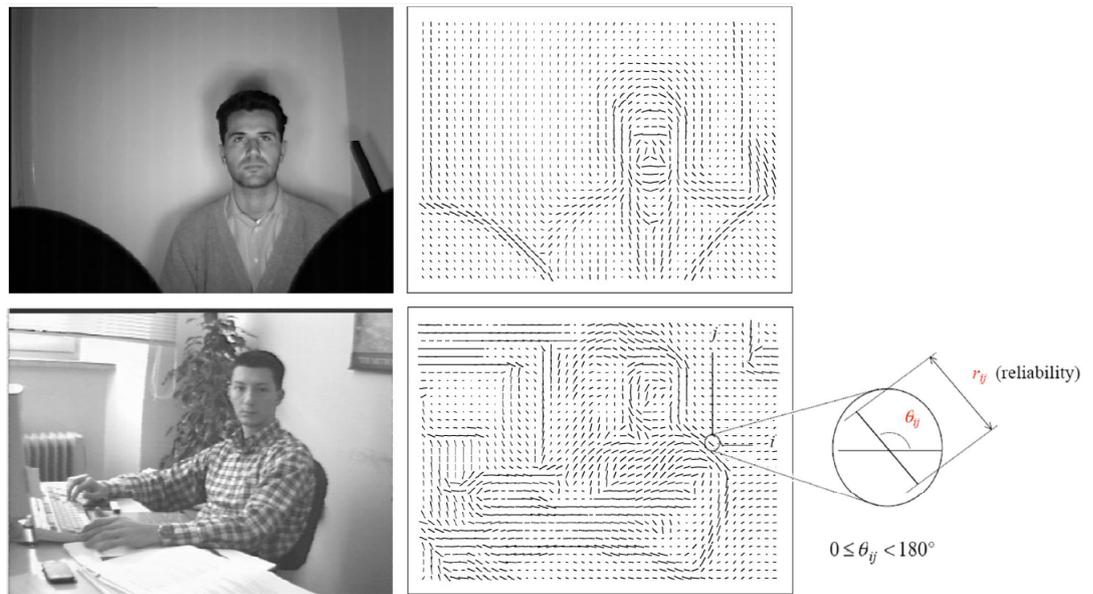


Figura 5.8: Local Orientation Histogram: risultato del calcolo dell'immagine direzionale a partire da due generiche immagini

L'approccio più semplice e naturale per l'estrazione dell'orientazione locale è basato sul calcolo del gradiente dell'immagine. Questo concetto non è affatto nuovo, visto che è già stato menzionato nella descrizione della fase di pre-processing ed in particolare per l'attività di estrazione dei contorni. Come sappiamo, il gradiente  $\nabla(x_i, y_j)$  nel punto  $[x_i, y_j]$  di una generica immagine  $I$  è un vettore bidimensionale  $[\nabla_x(x_i, y_j), \nabla_y(x_i, y_j)]$ , dove  $\nabla_x$  e  $\nabla_y$  sono rispettivamente le derivate di  $I$  in  $[x_i, y_j]$  rispetto alle direzioni  $x$  e  $y$  [44]. L'angolo di fase del gradiente indica la direzione del massimo cambiamento d'intensità dei pixel e la direzione  $\theta_{ij}$  di un ipotetico edge che attraversa la regione centrata in  $[x_i, y_j]$  è ortogonale all'angolo di fase del gradiente in  $[x_i, y_j]$ .

L'utilizzo del gradiente non è però sufficiente per il calcolo dell'immagine direzionale, per i seguenti motivi:

- non linearità e discontinuità attorno a  $90^\circ$ ;

- la stima di una singola orientazione rappresenta un'analisi di basso livello eccessivamente sensibile al rumore. D'altra parte non è possibile fare una semplice media di più gradienti a causa della circolarità degli angoli;
- il concetto di orientazione media non sempre è ben definito; ad esempio la media aritmetica tra  $5^\circ$  e  $175^\circ$  produrrebbe un angolo di  $90^\circ$ , mentre il valor medio più adeguato è  $0^\circ$ .

Una soluzione semplice ed elegante per superare il problema della circolarità consiste nel raddoppiare gli angoli; ogni elemento dell'immagine direzionale  $D$  è codificato allora dal seguente vettore:

$$d_{ij} = [r_{ij} \cdot \cos 2\theta_{ij}, r_{ij} \cdot \sin 2\theta_{ij}]$$

dove  $r_{ij}$  denota la coerenza (modulo) del vettore avente orientazione  $\theta_{ij}$ . L'immagine direzionale viene calcolata procedendo ad una suddivisione dell'immagine di input in finestre di dimensioni  $n \times n$ , il cui valore rappresenta uno dei parametri da impostare durante l'utilizzo dell'applicazione. In virtù di ciò, il singolo elemento dell'immagine direzionale è in realtà ottenuto come media delle orientazioni presenti localmente all'interno della finestra presa in esame. Tale media può essere calcolata effettuando separatamente la media per le due componenti  $x$  e  $y$ :

$$\bar{\mathbf{d}} = \left[ \frac{1}{n^2} \sum_{i,j} r_{ij} \cdot \cos 2\theta_{ij}, \frac{1}{n^2} \sum_{i,j} r_{ij} \cdot \sin 2\theta_{ij} \right]$$

Per completare il calcolo dell'immagine direzionale è necessario determinare il valore della cosiddetta orientazione dominante in ciascuna sottofinestra. L'orientazione locale dominante può essere calcolata in modo robusto tramite la seguente formula, basata sulla media locale delle stime del gradiente in una finestra centrata nel punto  $[x_i, y_j]$  e di dimensioni  $n \times n$  (ad esempio supponiamo  $n = 17$ ):

$$\begin{aligned} \theta_{ij} &= 90^\circ + \frac{1}{2} \text{atan2}(2G_{xy}, G_{xx} - G_{yy}), \\ G_{xy} &= \sum_{h=-8}^8 \sum_{k=-8}^8 \nabla_x(x_i + h, y_j + k) \cdot \nabla_y(x_i + h, y_j + k), \\ G_{xx} &= \sum_{h=-8}^8 \sum_{k=-8}^8 \nabla_x(x_i + h, y_j + k)^2, \\ G_{yy} &= \sum_{h=-8}^8 \sum_{k=-8}^8 \nabla_y(x_i + h, y_j + k)^2 \end{aligned}$$

$\nabla_x$  e  $\nabla_y$  sono le componenti  $x$  e  $y$  del gradiente calcolato su una maschera di Sobel  $3 \times 3$  e  $\text{atan2}(y, x)$  è l'arcotangente delle due variabili  $y$  e  $x$ . La coerenza

$r_{ij}$  di ciascuna orientazione è infine data da:

$$r_{ij} = \frac{\sqrt{(G_{xx} - G_{yy})^2 + 4G_{xy}^2}}{G_{xx} + G_{yy}}$$

Rispetto a quanto mostrato nella precedente Figura 5.8, il calcolo dell'immagine direzionale nella feature di forma Local Orientation Histogram non viene effettuata a partire dallo sketch o dalla foto originale del volto, bensì dalla consueta immagine grayscale dei contorni. In particolare, l'immagine dei contorni viene suddivisa in celle attraverso la costruzione di una matrice  $M \times N$  del tutto equivalente a quella vista per la feature Shape Matrix. Ogni cella individua una ben precisa porzione dell'immagine grayscale dei contorni e il calcolo dell'immagine direzionale viene eseguito sul contenuto di ogni cella, trattando ognuna come se fosse un'immagine a sè stante.

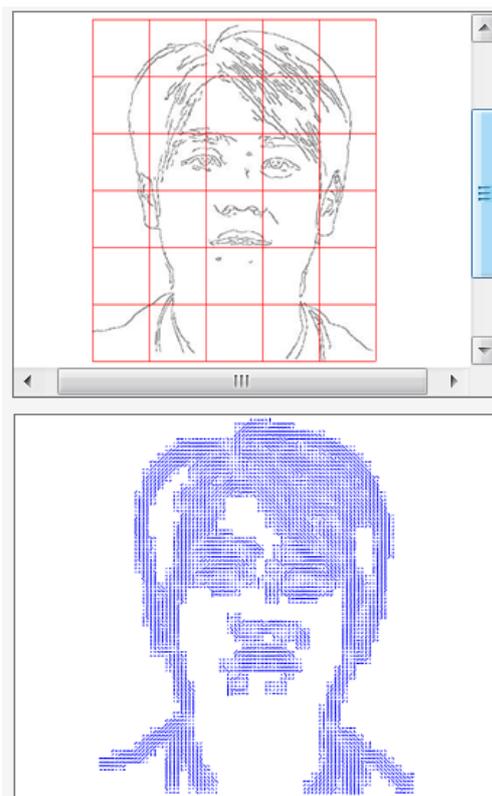


Figura 5.9: Local Orientation Histogram: immagine direzionale ottenuta a partire dall'immagine dei contorni. Su quest'ultima viene sovrapposta una matrice  $M \times N$  (in rosso) e l'immagine direzionale complessiva è ottenuta come composizione delle immagini direzionali calcolate su ogni cella separatamente dalle altre

Ricomponendo poi i risultati parziali delle immagini direzionali ottenute da ciascuna cella, si ottiene quanto rappresentato in basso in Figura 5.9: l'immagine dei contorni risulta trasformata in una nuova immagine del tutto affine a quella già vista in precedenza. L'unica differenza è che in questo caso l'immagine direzionale viene calcolata a partire da qualcosa che è a sua volta frutto dell'operazione di estrazione dei contorni.

Il calcolo dell'immagine direzionale fino a qui illustrato non è che il primo passo di elaborazione per l'estrazione della feature Local Orientation Histogram. In particolare, il calcolo dell'immagine direzionale è preparatorio per la successiva attività di costruzione degli istogrammi caratteristici di ogni cella della matrice. Infatti, all'interno di ogni cella sono collocati i vari elementi dell'immagine direzionale caratteristici della porzione di contorni presa in esame; l'idea è a questo punto quella di calcolare, per ogni cella, un istogramma delle orientazioni localmente presenti all'intero di essa. Nello specifico, ogni bin dell'istogramma sarà rappresentativo del numero di elementi che sono caratterizzati da un certo valore di orientazione. Siccome le orientazioni possono originariamente assumere un qualsiasi valore compreso tra  $0^\circ$  e  $180^\circ$ , si rende innanzi tutto necessaria una discretizzazione di questo range in  $Z$  intervalli, con  $Z$  parametro dell'applicazione. A questo punto, sarà sufficiente verificare in quale intervallo discreto ricade ogni orientazione  $\theta_{ij}$  contenuta all'interno di una cella: se essa ricade all'interno del  $k$ -esimo intervallo, ad essere incrementato di una unità sarà il  $k$ -esimo bin dell'istogramma associato alla cella. Ripetendo questa procedura elemento per elemento all'interno di ogni singola cella, ci ritroveremo con una serie di  $M * N$  istogrammi ognuno caratterizzato da  $Z$  bin e rappresentativo delle orientazioni localmente presenti nella cella stessa: questo insieme di istogrammi costituirà il descrittore per la feature in esame.

A conclusione di questo paragrafo non rimane che specificare la metodologia di confronto adottata per valutare la distanza tra due descrittori calcolati a partire da uno sketch e a partire da una foto. Definito con  $Hsketch_{(u,v)}$  l'istogramma relativo alla cella di riga  $u$  e colonna  $v$  per uno sketch e con  $Hfoto_{(u,v)}$  l'istogramma relativo alla cella di riga  $u$  e colonna  $v$  per una foto, la misura di distanza complessiva tra sketch e foto è ottenuta come:

$$\sum_{u=1}^M \sum_{v=1}^N distance(Hsketch_{(u,v)}, Hfoto_{(u,v)})$$

Per calcolare la singola funzione *distance* si possono riutilizzare a tutti gli effetti i criteri di distanza tra istogrammi descritti per la feature Gray Level Profile.

### 5.1.4 Fourier Descriptors

La feature Fourier Descriptors [28] presenta un importante segno di discontinuità rispetto alle feature di forma analizzate nei paragrafi precedenti; fino a questo momento, infatti, i descrittori erano collocati concettualmente nel medesimo dominio dell'immagine, cioè il dominio spaziale. Con questa nuova feature andiamo invece a prendere in esame descrittori definiti all'interno di un nuovo dominio chiamato appunto dominio di Fourier o dominio delle frequenze. L'utilizzo dello spazio di Fourier fornisce una valida alternativa ai metodi convenzionali, in quanto la rappresentazione di una forma in questo spazio consente di far emergere caratteristiche e apparenze che difficilmente potrebbero essere evidenziate rimanendo nel dominio spaziale. Numerosi sono gli studi effettuati nel tentativo di applicare questo tipo di feature per la classificazione e il riconoscimento di forme; per comprenderne meglio il significato, nel prosieguo del paragrafo vedremo innanzi tutto le caratteristiche di Fourier Descriptors in applicazione a forme generiche come previsto ad esempio da Zhang e Lu in [45] e [46] mentre nell'ultima parte vedremo le varianti applicate per rendere questa feature ancor più adatta al confronto sketch - foto.

Nella sua versione originale utilizzata dai sopra citati Zhang e Lu, la feature Fourier Descriptors è ottenuta applicando la trasformata di Fourier alla cosiddetta *shape signature*, cioè una funzione monodimensionale ricavata a partire dalle coordinate dei punti di contorno della figura stessa. Il nome *shape signature* è traducibile in italiano col termine di 'firma della figura': esattamente come la firma autografa si rivela un segno fortemente identificativo per l'individuo, la *shape signature* deve essere particolarmente rappresentativa del contorno della figura. Calcolando la trasformata di Fourier di tale funzione si ottengono dei coefficienti che, opportunamente normalizzati, prendono appunto il nome di descrittori di Fourier. Il motivo per il quale risulta interessante effettuare la trasformata della *shape signature* passando dal dominio spaziale al dominio di Fourier è che in questo dominio è sufficiente considerare i soli coefficienti delle basse frequenze per dare buona approssimazione della forma della figura in questione, mentre i coefficienti delle alte frequenze risultano essere di dettaglio e quindi trascurabili. Da ciò risulta evidente che la scelta della *shape signature* è determinante, visto che il descrittore complessivo sarà ottenuto come una sua rappresentazione nel dominio di Fourier alle basse frequenze: coordinate complesse, funzione di curvatura, funzione cumulativa angolare e distanza dal centro sono le *shape signature* più utilizzate nelle applicazioni pratiche di questa feature [45].

Entrando maggiormente nel dettaglio, considerando con  $r(t)$  la funzione *shape signature* descrivente il contorno della forma, la trasformata di Fourier discreta di  $r(t)$  genera tanti valori complessi  $a_n$  quanti sono i punti di contorno scelti per la costruzione della funzione monodimensionale stessa. Ogni

coefficiente  $a_n$  è calcolato come:

$$a_n = \frac{1}{N} \sum_{t=0}^{N-1} r(t) \exp\left(\frac{-j2\pi nt}{N}\right) \quad n = 0, 1, \dots, N-1$$

In molte applicazioni pratiche la funzione monodimensionale  $r(t)$  utilizzata è la cosiddetta *centroid distance*, cioè la distanza dei punti di bordo dal centroide  $(x_c, y_c)$  della figura. Dopo aver rilevato le coordinate  $(x(t), y(t))$  degli  $N$  punti di bordo della figura in esame, la funzione centroid distance può essere definita nel modo seguente:

$$r(t) = [(x(t) - x_c)^2 + (y(t) - y_c)^2]^{\frac{1}{2}}, \quad t = 0, 1, \dots, N-1$$

dove:

$$x_c = \frac{1}{N} \sum_{t=0}^{N-1} x(t), \quad y_c = \frac{1}{N} \sum_{t=0}^{N-1} y(t)$$

In Figura 5.10 è possibile visualizzare un primo esempio di calcolo della funzione centroid distance: dall'oggetto, che in questo caso assume le sembianze di una mela, vengono campionati alcuni punti di bordo e a partire da essi si calcolano le distanze con il centroide. Figura 5.10 ci mostra anche una rappresentazione cartesiana della distanza centroide ottenuta mettendo in sequenza le distanze calcolate.

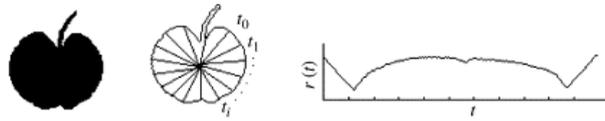


Figura 5.10: Fourier Descriptors: esempio della funzione centroid distance calcolata su una generica forma

La funzione distanza calcolata sul centroide risulta invariante solo alle rotazioni e alle traslazioni e di conseguenza i descrittori di Fourier ottenuti dall'operazione di trasformazione devono essere normalizzati, in modo tale che essi non dipendano dal fattore di scala utilizzato e dal punto di contorno iniziale preso in considerazione. Nello specifico, ogni descrittore di Fourier normalizzato  $b_n$  è ottenuto come:

$$b_n = \frac{a_n}{a_0}$$

dove  $a_0$  è il primo della serie di descrittori ottenuti mediante trasformata discreta di Fourier della funzione monodimensionale  $r(t)$ . Dalla teoria di Fourier risulta inoltre che  $b_n$  sia un numero complesso e quindi contraddistinto da una duplice informazione, una relativa al modulo ed una relativa alla fase del

descrittore stesso. L'informazione di fase non viene considerata e vengono considerati i soli valori di modulo: ciò che ne risulta è che sono sufficienti pochi coefficienti relativi alle basse frequenze per descrivere la forma. Ad esempio, in Figura 5.11, possiamo notare le funzioni centroidi di 3 forme del tutto simili alla mela di Figura 5.10 ma con piccole variazioni l'una rispetto all'altra, mentre in Figura 5.12 sono riportati i risultati di ricostruzione delle 3 forme mediante trasformata inversa a partire dai soli coefficienti alle basse frequenze. In particolare, in Figura 5.12 (a) la ricostruzione è il risultato della trasformata inversa a partire dai primi 3 descrittori di Fourier, mentre in Figura 5.12 (b) è riportato il risultato della ricostruzione a partire dai primi 5 descrittori. Il punto di forza dei descrittori di Fourier risulta evidente: per descrivere una forma nel dominio spaziale mediante funzione centroid distance si rende necessario campionare molteplici punti sul bordo, mentre per descriverla con una buona approssimazione nel dominio di Fourier è sufficiente un numero notevolmente inferiore di coefficienti.

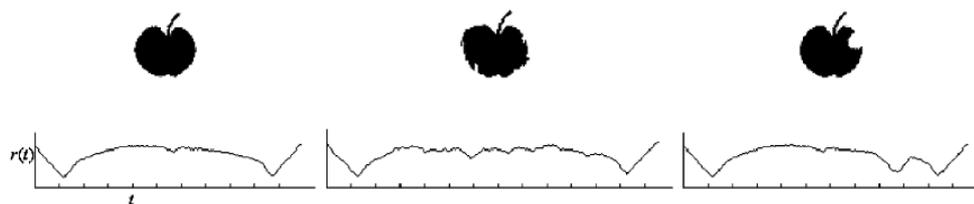


Figura 5.11: Fourier Descriptors: funzione centroid distance calcolata su 3 forme distinte ma simili



Figura 5.12: Fourier Descriptors: ricostruzione delle 3 forme di Figura 5.11 a partire dai primi 3 descrittori di Fourier (a) e a partire dai primi 5 descrittori di Fourier (b) ottenuti dal processo di trasformata delle rispettive funzioni centroid distance

Un altro esempio può essere visualizzato in Figura 5.13 e riguarda l'applicazione dei descrittori di Fourier su una forma particolare come la lettera E. In particolare, si suppone in questo caso di campionare  $N = 1024$  punti del bordo e di ottenere quindi 1024 descrittori di Fourier dal processo di trasformazione. Ricostruendo la forma a partire da tutti e 1024 i descrittori mediante antitrasformata si riottiene la forma di partenza, ma lo stesso risultato può essere ottenuto con un'ottima approssimazione anche considerandone molti di meno,

come ad esempio i primi 201. Questo ribadisce che la vera informazione è contenuta alle basse frequenze, mentre alle alte frequenze sono presenti coefficienti di dettaglio che possono anche essere non considerati.

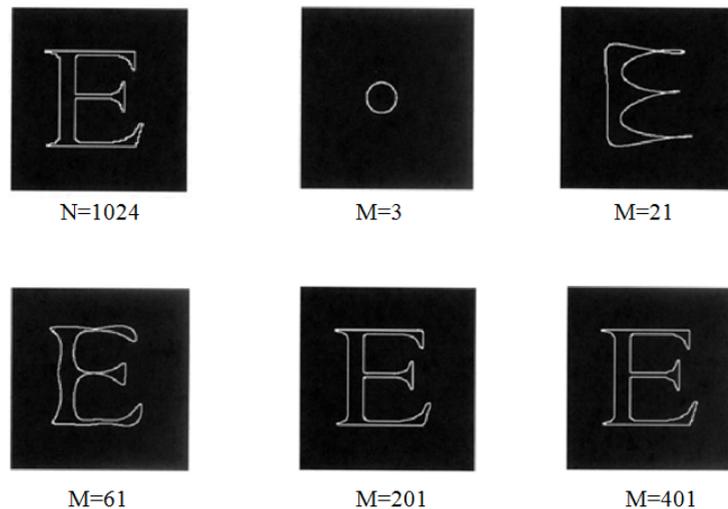


Figura 5.13: Fourier Descriptors: diversi casi di ricostruzione di una forma a partire dai primi  $M$  descrittori di Fourier

I principi illustrati fino a questo momento che abbiamo visto applicati a forme generiche sono stati presi in prestito per la tematica di nostro interesse legata al riconoscimento del volto. Ci sono però due importanti differenze nel calcolo della feature Fourier Descriptors rispetto a quanto detto in precedenza:

- la shape signature del volto viene calcolata in modo simile a quanto visto per la funzione centroid distance, con la differenza che il centroide  $(x_c, y_c)$  è predeterminato come punto medio tra i due occhi e le singole distanze euclidee tra centroide e punti di bordo vengono normalizzate rispetto alla distanza euclidea che separa i due occhi. In particolare:

$$r(t) = \frac{[(x(t) - x_c)^2 + (y(t) - y_c)^2]^{\frac{1}{2}}}{EyeDistance}, \quad t = 0, 1, \dots, N - 1$$

Come già visto nella feature Gray Level Profile, è piuttosto banale ricavare distanza e punto medio degli occhi, visto che le informazioni relative ai centri delle due pupille è già a nostra disposizione dalla fase di pre-processing. Per quanto riguarda il campionamento dei punti di bordo, si definisce uno *step-size* verticale e in base a questo si individuano i punti esterno del contorno del volto: in Figura 5.14 si può notare che definendo uno step size abbastanza ridotto il campionamento effettuato è piuttosto fitto individuando un numero elevato di punti (crocette nere);

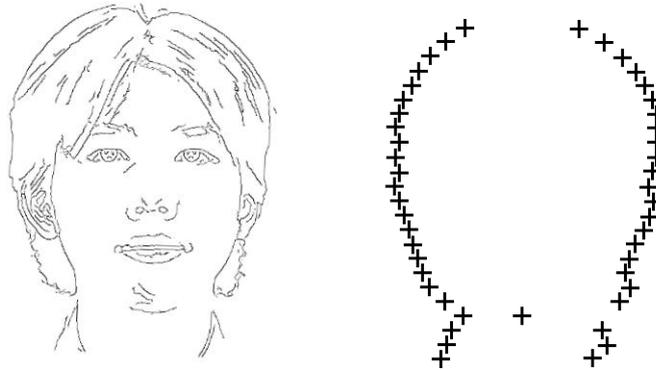


Figura 5.14: Fourier Descriptors: campionamento dei punti esterni di bordo del volto per la costruzione della shape signature

- piuttosto che utilizzare la trasformata di Fourier discreta si è preferito puntare sulla cosiddetta DCT (*Cosine Discrete Transform*). Il motivo è che la trasformata DCT è molto utilizzata in quegli ambiti dove è di fondamentale importanza che l'informazione significativa sia concentrata nelle basse frequenze, come ad esempio gli algoritmi di compressione di immagini. Inoltre, essendo la DCT basata su uno sviluppo di Fourier fatto di soli coseni, risulta più 'leggera' a livello computazionale rispetto alla classica trasformata di Fourier discreta. La formula di trasformazione utilizzata che permette di ottenere i descrittori desiderati è la seguente e corrisponde alla versione DCT-II:

$$a_n = \sum_{t=0}^{N-1} r(t) \cos \left[ \frac{\pi}{N} \left( t + \frac{1}{2} \right) n \right] \quad n = 0, 1, \dots, N - 1$$

I descrittori ottenuti sono poi sottoposti un processo di normalizzazione del tutto analogo a quello già illustrato in precedenza.

Il risultato del processo di calcolo dei descrittori di Fourier può essere visualizzato in Figura 5.15 e in Figura 5.16. Nella prima abbiamo una rappresentazione grafica dei primi 20 descrittori di Fourier per uno sketch ed una foto relativi allo stesso individuo, mentre nel secondo caso sketch e foto sono afferenti a soggetti distinti. Come si nota, i primi 20 di una lunga serie di descrittori (nelle due Figure i descrittori totali per ciascun volto sono oltre 130) sono già sufficienti per capire se i volti confrontati abbiano o meno la stessa forma. Il lettore può notare come l'andamento dei descrittori riportato nel grafico cartesiano sia piuttosto simile anche nel caso di volti distinti, ma la presenza di piccole differenze locali consente di intuire la diversità delle forme a cui fanno riferimento. Per dare misura oggettiva della diversità tra volti per mezzo dei

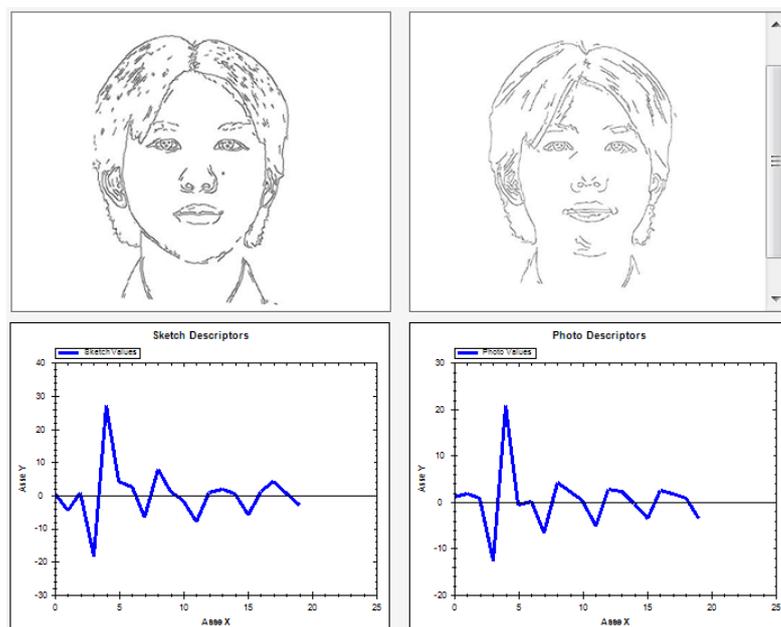


Figura 5.15: Fourier Descriptors: descrittori alle basse frequenze per uno sketch ed una foto relativi al medesimo soggetto

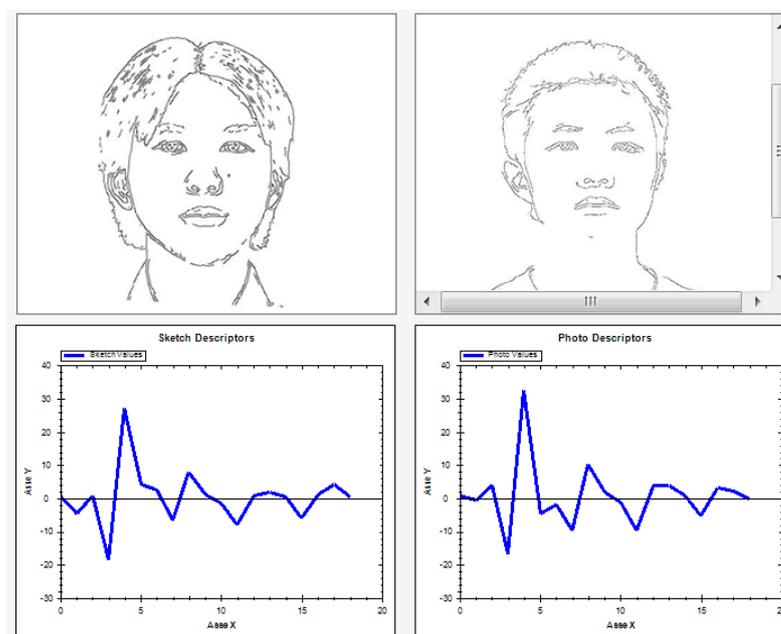


Figura 5.16: Fourier Descriptors: descrittori alle basse frequenze per uno sketch ed una foto relativi a soggetti distinti

loro descrittori di Fourier, è sufficiente utilizzare la distanza euclidea tra di essi considerando ovviamente i soli descrittori alle basse frequenze.

### 5.1.5 Local Binary Pattern

La feature presa in esame in questo paragrafo prende il nome di Local Binary Pattern (LBP) e, seppur inserita in questa tesi nel contesto delle cosiddette feature di forma, nasce in realtà con l'obiettivo di fornire un valido descrittore per la texture dell'oggetto rappresentato all'interno di un'immagine. Grazie al proprio potere discriminativo, alla sua semplicità computazionale, oltre che alla robustezza rispetto alle variazioni di illuminazione, l'operatore LBP è ben presto diventato un approccio molto popolare in svariate applicazioni come ad esempio il riconoscimento facciale 2D a partire da immagini grayscale del volto. Il nostro interesse è quello di valutare l'efficacia di questa feature quando l'immagine grayscale in input è in realtà quella relativa ai soli contorni del volto, esattamente come accade per tutte le altre feature di forma analizzate fino a questo momento.

L'operatore LBP è stato introdotto da Ojala *et al.* [47] per l'analisi della già menzionata tessitura delle immagini. Nella sua versione base il funzionamento dell'operatore è molto semplice: a ciascun pixel dell'immagine viene assegnato un valore binario dipendente dalla valorizzazione dei pixel che stanno nell'intorno  $3 \times 3$  di quello preso in esame. Sia ad esempio  $p$  un pixel dell'intorno e sia  $p_c$  il pixel centrale in una situazione come quella rappresentata in Figura 5.17: se  $p$  ha un valore superiore o uguale a quello di  $p_c$  allora a  $p$  è assegnato il valore 1, altrimenti il valore 0. Mettendo in sequenza i valori binari ottenuti confrontando il pixel centrale con ciascun pixel dell'intorno si ottiene un numero binario di 8 cifre: la sua conversione in sistema decimale rappresenta il valore finale associato al pixel  $p$  preso come riferimento. Questa versione base di LBP ha un inconveniente legato alla dimensione dell'intorno; essa infatti ha un valore prefissato pari a 3 mentre potrebbe essere preferibile utilizzare intorni di dimensione diversa a seconda del caso applicativo specifico. Per questo motivo, l'operatore LBP è stato esteso per gestire intorni di dimensione variabile. In particolare, nella nuova versione, si passa dal considerare un intorno quadrato al considerare un intorno circolare del pixel introducendo il concetto di interpolazione per quei punti campione dell'intorno che non cadono esattamente su un pixel dell'immagine. Come indicato anche in Figura 5.18, l'operatore LBP prevede in questo caso la definizione di due nuovi parametri: il numero di punti campione  $P$  e il raggio dell'intorno circolare  $R$ . Considerare intorni circolari e utilizzare l'interpolazione per dare un preciso valore a ciascun punto campione dell'intorno non cambia il processo di costruzione del valore binario associato al pixel centrale. Quanto illustrato per l'intorno quadrato  $3 \times 3$  vale anche qualora si utilizzi un intorno circolare; unitamente al fatto che l'intorno circolare prevede un numero piuttosto elevato di punti campione, l'effetto che

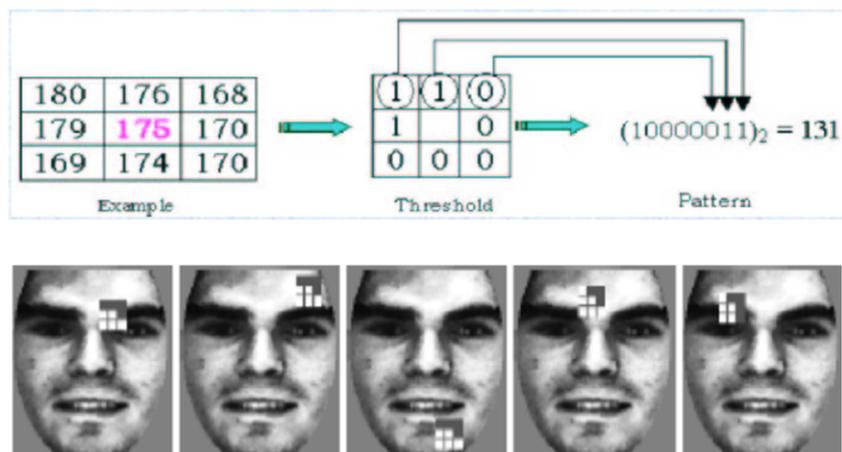


Figura 5.17: Local Binary Pattern: calcolo dell'operatore LBP su un'immagine grayscale del volto del soggetto

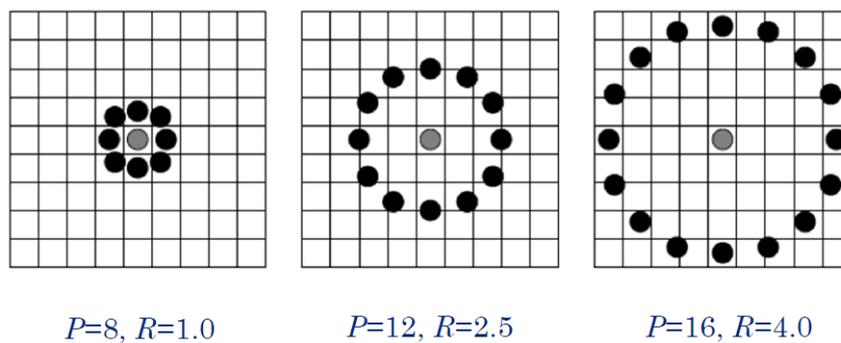


Figura 5.18: Local Binary Pattern: intorno circolare di un pixel per il calcolo di LBP

si ottiene è la produzione di particolari pattern binari contraddistinti dal numero e dalla sequenza di transizioni 0-1. In Figura 5.19 possiamo visualizzare alcuni esempi di pattern binari uniformi, chiamati in questo modo per la loro particolarità di rappresentare alcune fra le strutture locali più rilevanti come edge, spot, ecc..

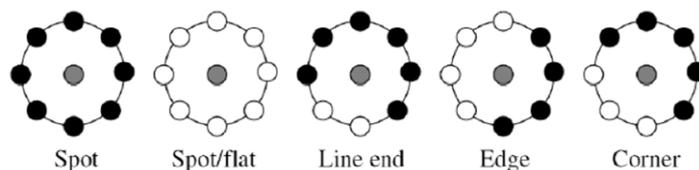


Figura 5.19: Local Binary Pattern: esempi di strutture di pattern uniformi

Nello specifico, un pattern è detto uniforme quando, considerato in modo circolare, contiene al massimo due transizioni 0-1 o 1-0 (ad esempio i pattern 10000011, 11110000, 00000000 sono uniformi). Il motivo per cui risulta conveniente considerare solo i pattern uniformi risiede nel risparmio di memoria, in quanto i pattern totali sono  $2^P$  mentre i pattern uniformi sono solo  $P(P-1)+2$ . Per capire meglio questo concetto possiamo ad esempio considerare Figura 5.20: l'immagine originale (a sinistra) viene rappresentata nella versione che evidenzia con puntini bianchi i pixel con pattern uniforme (immagine al centro) e nella versione che, in modo analogo, evidenzia i pixel con pattern non uniforme (immagine a destra). Si può pertanto notare come la prima contenga molti meno pixel bianchi rispetto a quest'ultima, a conferma di quanto illustrato in precedenza.



Figura 5.20: Local Binary Pattern: pattern uniformi e non uniformi individuati all'interno di un'immagine grayscale del volto

Calcolare LBP sull'intera immagine significa produrre come risultato di output un vettore di feature consistente in un istogramma (eventualmente normalizzato). Il vettore viene ottenuto dividendo innanzi tutto l'immagine in  $k^2$  sottofinestre e per ogni sottofinestra è costruito un istogramma in cui a ciascuno bin è associato a uno specifico pattern binario. I bin sono quindi in totale pari a  $P(P-1)+3$ :  $P(P-1)$  bin per i pattern con 2 transizioni, 2 bin per i pattern con 0 transizioni ed un solo bin per i pattern non uniformi. L'ultimo passo

consiste nel normalizzare e concatenare gli istogrammi calcolati per tutte le sottofinestre, come mostra Figura 5.21.

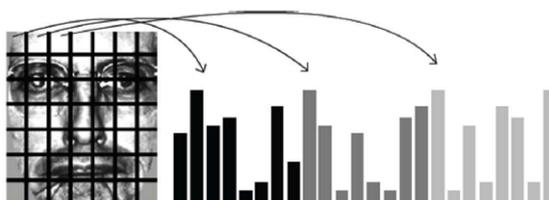


Figura 5.21: Local Binary Pattern: istogrammi costituenti il vettore di feature

Come il lettore avrà notato, gli esempi mostrati per illustrare il funzionamento di LBP fanno riferimento ad immagini grayscale del volto. Nel nostro caso LBP viene invece applicato direttamente ad immagini grayscale contenenti i soli contorni dei volti, in conformità con quanto accade per tutte le altre feature di forma. Essendo LBP molto sensibile alla tessitura dell'immagine e alla eventuale presenza di pixel di 'rumore', si rende necessaria l'introduzione di un nuovo parametro, oltre ai già citati numero di punti campione e raggio dell'intorno circolare, detto soglia grayscale. Questo parametro serve per scartare pixel dell'immagine grayscale che, pur non assumendo una tonalità esattamente bianca e non uguale in valore a 255, si avvicinano a tale colorazione e possono essere considerati ugualmente come sfondo. Ad esempio, fissando questa soglia ad un valore pari a 250, si assume che i pixel che hanno un valore grayscale superiore a 250 sono da considerare comunque come pixel di sfondo. Si è verificato sperimentalmente che l'introduzione di tale soglia consente di migliorare l'efficacia di LBP per il riconoscimento.

Il risultato prodotto da LBP, come detto, consiste in un insieme di istogrammi concatenati fino a formare un vettore numerico; nel confrontare il vettore estratto dai contorni di uno sketch e quello estratto dai contorni di una foto, si possono teoricamente utilizzare le consuete misure di distanza di tra istogrammi. A livello pratico, però, avendo tali vettori un numero molto elevato di componenti a causa della concatenazione multipla di istogrammi, risulta più conveniente utilizzare la classica misura di distanza euclidea.

### 5.1.6 Beam Angle Statistics

La feature studiata in questo paragrafo prende il nome di Beam Angle Statistics e la sua utilità consiste nel fornire un valido metodo per la descrizione della forma del contorno esterno del volto. Beam Angle Statistics presenta, da questo punto di vista, una somiglianza con Fourier Descriptors visto che anche quest'ultima si basava sull'analisi del contorno esterno del volto. A differenza però della feature di forma analizzata nel paragrafo 5.1.4, con Beam Angle Statistics non si effettua alcuna trasformazione da un dominio all'altro; il dominio

nel quale si lavora per il calcolo di questa feature è il medesimo dominio in cui si colloca concettualmente l'immagine dei contorni del volto, cioè il dominio spaziale.

Beam Angle Statistics, ideata da Arica e Vural [48] per il riconoscimento di forme in maniera invariante rispetto a traslazioni, rotazioni, scala e rumore, si basa sul concetto di fascio (beam) di angoli originati a partire da un punto di bordo. In particolare, gli angoli generati sono quelli che insistono sui vertici creatisi mediante connessione tra il punto di bordo preso come riferimento e gli altri punti campione situati sul bordo della figura. Per comprendere meglio questo meccanismo, prendiamo in esame la forma in Figura 5.22 e i punti campione individuati sul proprio bordo (pallini neri).

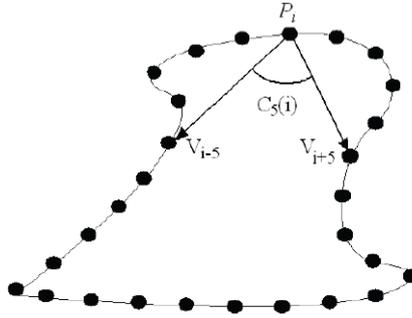


Figura 5.22: Beam Angle Statistics: esempio di calcolo dell'angolo di ordine  $k$  (con  $k = 5$ )

In particolare, indicando con  $B$  il bordo della forma, possiamo affermare che tale bordo può essere visto come una sequenza connessa di  $N$  punti  $P_i = (x_i, y_i)$ , cioè:

$$B = \{P_1, P_2, \dots, P_N\}$$

Tra ciascun punto  $P_i$  ed ogni altro punto del bordo è possibile tracciare un segmento che li congiunge; il segmento che unisce il punto  $P_i$  con il punto che si trova  $k$  posizioni più avanti è chiamato *forward beam vector* di ordine  $k$ :

$$V_{i+k} = \overrightarrow{P_i P_{i+k}}$$

In modo analogo, è definito come *backward beam vector* il segmento congiungente il punto  $P_i$  con il punto di bordo che risiede  $k$  posizioni più indietro:

$$V_{i-k} = \overrightarrow{P_i P_{i-k}}$$

Definiti i vettori  $V_{i+k}$  e  $V_{i-k}$ , il beam angle  $C_k(i)$  di ordine  $k$  che insiste sul punto  $P_i$  è ottenuto così:

$$C_k(i) = (\theta_{V_{i+k}} - \theta_{V_{i-k}})$$

dove:

$$\theta_{V_{i+k}} = \arctan\left(\frac{y_{i+k} - y_i}{x_{i+k} - x_i}\right) \quad \theta_{V_{i-k}} = \arctan\left(\frac{y_{i-k} - y_i}{x_{i-k} - x_i}\right)$$

Il lettore avrà sicuramente notato come nel nome di questa feature compaia un riferimento all'ambito della statistica; il motivo risiede nel fatto che per ciascun punto di bordo  $P_i$ , l'angolo  $C_k(i)$  può essere considerato come una variabile con funzione di densità di probabilità  $P(C_k(i))$ . Il momento  $m$ -esimo della variabile  $C_k(i)$  è così definito:

$$E[C_k^m(i)] = \sum_{k=1}^{\frac{N}{2}-1} C_k^m(i) \cdot P_k(C_k(i)) \quad m = 1, 2, \dots$$

dove  $E$  indica il valore atteso. In Figura 5.23 vengono riportate a titolo d'esempio le funzioni densità di probabilità di momento 1, 2 e 3 di due forme: quella a sinistra è la versione originale e quella a destra è la versione affetta da distorsioni. Si può notare come BAS riesca a catturare informazioni percettive e a dare una descrizione globalmente simile per le due forme con una buona invarianza rispetto alla presenza di rumore.

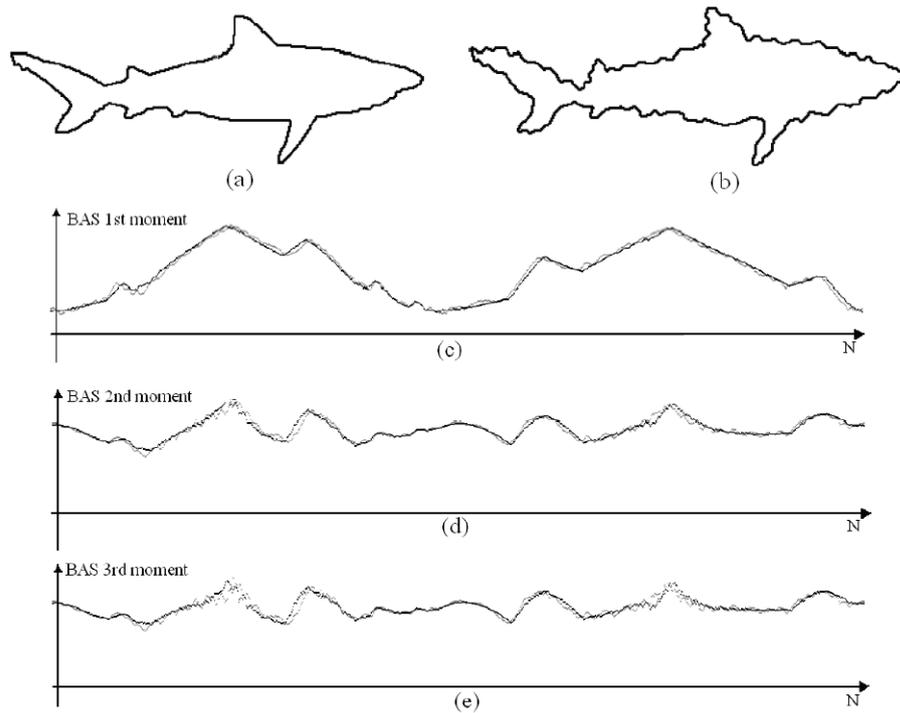


Figura 5.23: Beam Angle Statistics: contorno originale (a), contorno rumoroso (b), plot dei momenti BAS di ordine 1 (c), 2 (d), 3 (e)

Al di là delle considerazioni di tipo statistico, l'aspetto interessante di questa

feature è la capacità di poter descrivere il contorno esterno per mezzo dei cosiddetti momenti di ordine  $k$ , calcolando il valore degli angoli individuati dai punti di bordo nel modo precedentemente illustrato. Pertanto, l'applicazione pratica di BAS sulle nostre immagini contenenti i soli contorni prevede innanzi tutto di individuare una serie di punti campione situati sul bordo esterno del volto stesso. Questo campionamento avviene in maniera analoga a quanto visto per la feature Fourier Descriptors, con la differenza che al posto di definire uno step size si definisce direttamente il numero  $N$  di punti campione da individuare. Successivamente, per ogni punto campione  $P_i$  selezionato sul bordo della figura, vengono calcolati gli angoli che esso forma con i  $k$  punti che seguono e con i  $k$  punti che lo precedono. Questi angoli vengono poi sottratti rispettivamente tra loro ottenendo come risultato  $k$  valori angolari corrispondenti ai momenti di ordine  $1, 2, \dots, k$  del punto  $P_i$ . Rilevare tutti i momenti a partire da quello di ordine 1 fino a quello di ordine  $k$  consente di rendere più robusto il descrittore finale, consistente in una serie di valori angolari ordinati per punto campione preso come riferimento. Per migliorare l'efficacia della feature, si è preferito lavorare indipendentemente sulle due metà del volto; il risultato è la produzione di un descrittore che consiste in realtà in una coppia di vettori, uno contenente i valori angolari relativi alla porzione sinistra del volto ed uno relativo alla porzione destra. A tal proposito, per evitare che discontinuità nei contorni potessero portare ad individuare punti campione non situati sul bordo esterno del volto ma collocati all'interno del viso, è stata prevista l'introduzione di una soglia per delimitare la regione non idonea ai fini della ricerca del punto campione. La regione non idonea viene ottenuta attraverso una riduzione percentuale del bounding box del volto. In Figura 5.24 è evidenziata la linea di delimitazione (in rosso) oltre che punti campionati sul bordo esterno del volto.

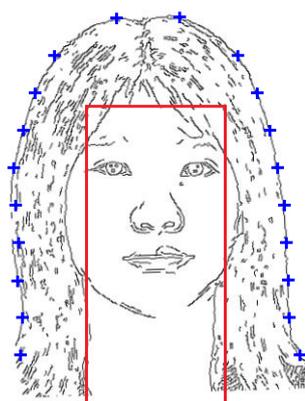


Figura 5.24: Beam Angle Statistics: punti campione (crocette blu) e linea di delimitazione (linea rossa) per il calcolo della feature sul contorno esterno del volto

Per quanto concerne infine il calcolo della similarità tra uno sketch ed una foto sulla base dei vettori calcolati sulle rispettive metà destre e sinistre, essa viene valutata facendo la semplice differenza aritmetica, componente per componente, dei vettori stessi. I risultati ottenuti a partire dalle singole metà vengono poi mediati, ottenendo un valore di similarità complessivo che esprime una valutazione oggettiva della somiglianza tra le forme dei volti.

### 5.1.7 Chain Code e Pixel Decimal Value

Le ultime due feature di forma che restano da analizzare prendono il nome di Chain Code e Pixel Decimal Value; pur essendo ben distinte l'una rispetto all'altra, esse hanno delle caratteristiche in comune ed è quindi conveniente trattarle assieme.

L'algoritmo alla base della feature Chain Code trae spunto dall'idea di Liu *et al.* [49], i quali hanno proposto una interessante feature chiamata Vertex Chain Code in grado di dare una descrizione locale dei pixel di foreground. Per capire meglio in cosa consiste l'intuizione alla base della nostra feature consideriamo un insieme di pixel contigui che formano una matrice  $4 \times 4$  come quella rappresentata in Figura 5.25.

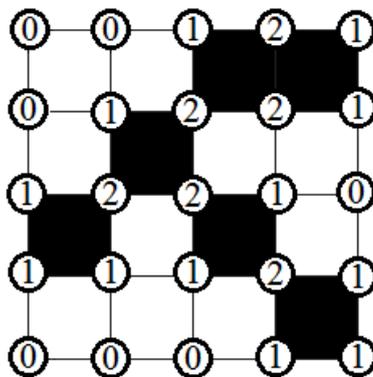


Figura 5.25: Chain Code: semplice esempio di calcolo delle etichette numeriche associate ai vertici dei pixel

I pixel di questa matrice assumono solo due possibili colorazioni: nera se il pixel è di foreground, bianca se di background. L'attenzione si deve a questo punto focalizzare non tanto sui singoli pixel, quanto sui vertici che insistono fra di essi: calcolare il Chain Code di questa matrice significa determinare, per ogni vertice della matrice, quanti sono i pixel di foreground che si trovano nel proprio intorno. L'etichetta numerica presente all'interno di ciascun circoletto identifica esattamente il numero di pixel di foreground situati nell'intorno. Va da sè che ogni vertice possa avere un valore associato compreso tra 0, qualora non ci siano pixel di foreground ad esso vicini, e 4. Determinando il valore delle etichette numeriche per tutti i vertici della matrice ed ordinando i valori per

riga, si ottiene un vettore che rappresenta proprio il descrittore Chain Code della matrice di pixel.

L'intuizione appena descritta deve essere applicata a partire dall'immagine contenente i soli contorni del volto e pertanto si rendono necessarie alcune operazioni aggiuntive. La prima considerazione da fare è che l'immagine dei contorni, come sappiamo, è di tipo grayscale e quindi i pixel possono assumere non solo tonalità bianche e nere ma anche colorazioni intermedie. Di base potremmo supporre di considerare come background solo i pixel esattamente pari in valore a 255; per dare però la possibilità di gestire meglio il calcolo della feature, si è preferito introdurre la medesima soglia grayscale prevista a suo tempo per la feature Local Binary Pattern. Con l'ausilio di questa soglia, solo i pixel con tonalità di grigio inferiore alla soglia sono considerati effettivamente come foreground.

La seconda considerazione riguarda invece la dimensione del vettore finale contenente le etichette numeriche precedentemente descritte. In particolare, i vettori ottenuti da una foto e da uno sketch hanno un numero differente di componenti in quanto le dimensioni del volto non sono esattamente le stesse. Siccome confrontare vettori di dimensioni diverse non è possibile, si rende necessaria un'operazione di normalizzazione delle immagini dei contorni in modo da renderle esattamente identiche in dimensioni. Pertanto, dopo aver calcolato il bounding box del volto previsto anche per le altre feature, l'immagine contenuta nel bounding box viene ridimensionata a valori prefissati, in modo da produrre vettori Chain Code associati perfettamente confrontabili. Le misure di similarità adottate per calcolare la somiglianza fra foto e sketch per mezzo dei loro descrittori Chain Code sono la già citata distanza di Hamming e la differenza fra vettori componente per componente.

Come anticipato all'inizio di questo paragrafo, l'altra feature da trattare prende il nome di Pixel Decimal Value e per certi versi presenta caratteristiche simili a quella appena illustrata. Il calcolo di questa feature prevede di fare scorrere su tutta l'immagine una maschera  $3 \times 3$  come quella rappresentata in Figura 5.26 (a).

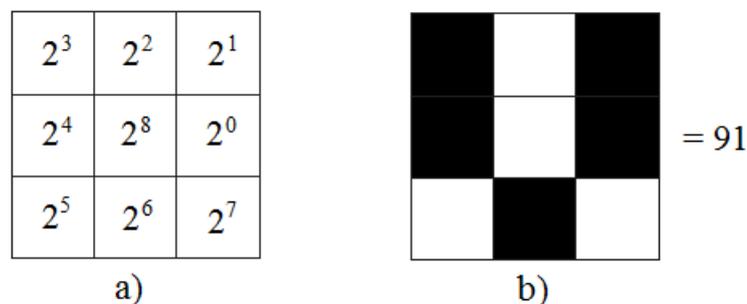


Figura 5.26: Pixel Decimal Value: maschera  $3 \times 3$  (a) da scorrere sull'immagine per il calcolo dei valori decimali (b)

Questa maschera indica quale valore associare a ciascun pixel dell'intorno compreso il pixel centrale, ed ogni valore è espresso sotto forma di potenza di 2. Una volta sovrapposta questa maschera su un certo pixel centrale  $p_c$ , si effettua la somma delle varie potenze in base 2 coinvolgendo però le sole potenze corrispondenti ad un pixel di foreground. In riferimento alla Figura 5.26 (b), il valore decimale risultante è ottenuto come:

$$2^0 + 2^1 + 2^3 + 2^4 + 2^6 = 91$$

A partire da un'immagine di dimensione  $w \times h$  e ripetendo questa operazione per ogni pixel dell'immagine con l'esclusione dei pixel di bordo per comodità, si ottiene un vettore di  $w * h$  valori numerici che costituisce il descrittore Pixel Decimal Value dell'immagine stessa. Anche per questa feature è prevista l'introduzione della soglia di grigio per la determinazione dei pixel di foreground e il ridimensionamento del bounding box per produrre vettori di pari dimensione. La misura di similarità utilizzata per il confronto tra descrittori sketch - foto è la sola distanza euclidea.

## 5.2 SURF (Speeded - Up Robust Features)

Le feature analizzate nella precedente sezione di questa relazione, che costituiscono il cuore del sistema di riconoscimento sketch - foto, possiedono un tratto distintivo che le accomuna: esse prendono in input immagini grayscale relative ai soli contorni del volto e prevedono il calcolo di descrittori numerici identificativi di aspetti legati alla forma e all'aspetto del volto. L'idea di convertire in via preliminare la foto e lo sketch del soggetto in immagini contenenti i soli edge del volto è sicuramente valida al fine di annullare il 'gap di modalità', ma potrebbe essere altrettanto interessante verificare la disponibilità di descrittori locali calcolabili e confrontabili direttamente sulle immagini originali. Per questo motivo, dopo aver trattato le feature di forma, in questo paragrafo prendiamo in esame un nuovo tipo di descrittore chiamato SURF che, come vedremo, rende possibile questo tipo di confronto diretto andando ad arricchire e a rendere più robusto il processo di riconoscimento.

Speeded - Up Robust Features è uno dei più recenti algoritmi di estrazione di feature da immagini ed è stato ideato da Bay *et al.* [29]; si tratta di una versione più efficiente e migliorata di un altro algoritmo piuttosto famoso chiamato SIFT (Scale Invariant Feature Transform) progettato da Lowe [30] qualche anno prima. SURF consente di estrarre feature distintive (cioè ben discriminabili l'una rispetto all'altra) all'interno di un'immagine, in modo invariante rispetto a rotazioni, traslazioni, cambiamenti di scala, occlusioni parziali, deformazioni e cambiamenti di illuminazione. Le feature vengono calcolate in corrispondenza di punti cosiddetti 'salienti', come ad esempio corner o edge, relativi agli oggetti presenti in un'immagine; il processo di estrazione è particolarmente

complesso e strutturato per garantire le proprietà di invarianza precedentemente menzionate. Le feature individuate consentono di dare una descrizione altamente robusta degli oggetti presenti in una scena, talmente robusta da permettere una successiva localizzazione dei medesimi oggetti anche a partire da scenari completamente differenti. A tal proposito, l'ideatore dell'algoritmo SURF ha curato sia l'aspetto legato all'individuazione delle feature, sia l'aspetto legato al matching tra feature estratte a partire da immagini diverse. Grazie alle proprietà menzionate, questo estrattore trova larga applicazione in svariati ambiti della Visione Artificiale, come il riconoscimento di oggetti, la visione 3D e video tracking.

Nel contesto di questa tesi, si vuole tentare l'utilizzo di questo algoritmo per il confronto sketch - foto. In particolare, per quanto concerne l'attività di estrazione delle feature, faremo completo riferimento alle caratteristiche dell'algoritmo SURF e nel prosieguo vedremo una panoramica di questo processo. In riferimento invece al matching tra punti estratti da una foto e da uno sketch useremo tecniche derivate in particolare dall'algoritmo SIFT ma proprie anche di SURF.

L'algoritmo SURF esegue la ricerca dei punti d'interesse a partire da una rappresentazione alternativa dell'immagine da processare, chiamata immagine integrale.

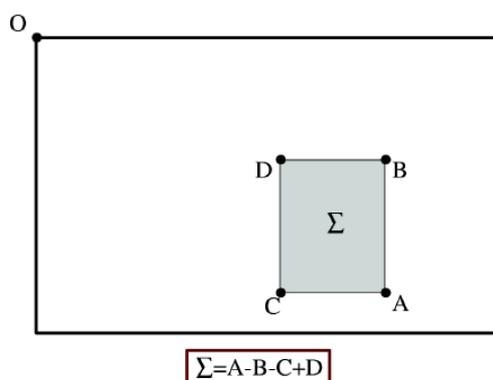


Figura 5.27: SURF: esempio di calcolo di una porzione dell'immagine integrale

Questa immagine, il cui calcolo non è affatto complesso, consente di velocizzare le successive operazioni di convoluzione. In particolare, data un'immagine  $I$  e un punto di coordinate  $(x, y)$ , l'immagine integrale  $I_{\Sigma}(x, y)$  è ottenuta come somma dei valori dell'intensità dei pixel compresi tra il punto in questione e l'origine  $O$ , posta nell'angolo superiore sinistro dell'immagine stessa. Cioè:

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

Con l'ausilio dell'immagine integrale la somma delle intensità di una qualsiasi porzione dell'immagine risulta notevolmente semplificata. A partire da un rettangolo di vertici A, B, C e D (Figura 5.27) che sottintendono valori integrali corrispondenti alle coordinate del relativo vertice, la somma dell'intensità dei suoi pixel è uguale a:

$$\sum = A + D - (C + B)$$

Le operazioni di somma sono alla base delle successive operazioni di convoluzione e l'utilizzo dell'immagine integrale permette un notevole risparmio di tempo calcolo.

La rilevazione dei punti d'interesse si basa principalmente sul determinante della matrice Hessiana per le sue proprietà di accuratezza, con l'obiettivo di localizzare punti e/o regioni dell'immagine aventi un contenuto più chiaro o più scuro rispetto ai pixel circostanti. Tipicamente, siamo abituati a intendere tali punti come pixel di bordo ma nel caso di SURF queste strutture prendono il nome di *blob feature* ed è proprio in queste posizioni che il determinante della matrice Hessiana assume valori positivi molto elevati. A livello pratico, l'algoritmo SURF prevede l'utilizzo di un filtro Gaussiano del secondo ordine normalizzato, che permette un'analisi spaziale e su diversi fattori di scala. Dato un punto  $\mathbf{x} = (x, y)$  di un'immagine I, la matrice Hessiana  $\mathcal{H}(\mathbf{x}, \sigma)$  in  $\mathbf{x}$  alla scala fissata  $\sigma$  è definita come segue:

$$\mathcal{H}(\mathbf{x}, \sigma) = \begin{bmatrix} L_{xx}(\mathbf{x}, \sigma) & L_{xy}(\mathbf{x}, \sigma) \\ L_{xy}(\mathbf{x}, \sigma) & L_{yy}(\mathbf{x}, \sigma) \end{bmatrix}$$

dove  $L_{xx}(\mathbf{x}, \sigma)$ ,  $L_{xy}(\mathbf{x}, \sigma)$  e  $L_{yy}(\mathbf{x}, \sigma)$  sono convoluzioni della derivata di secondo ordine della Gaussiana con l'immagine I nel punto  $\mathbf{x}$ . Le Gaussiane sono molto utili ai fini dell'analisi su diversi fattori di scala, ma a livello pratico si rende necessaria una loro discretizzazione e approssimazione. Questo porta ad una diminuzione di precisione, la quale, tuttavia, risulta abbondantemente compensata dall'efficienza computazionale con cui vengono svolte le convoluzioni grazie all'immagine integrale.

Come detto in precedenza, il determinante approssimato dell'Hessiana rappresenta la risposta del blob nella posizione  $\mathbf{x}$  dell'immagine. SURF prevede tra i suoi punti di forza l'invarianza rispetto alla scala ed è quindi necessario che i punti d'interesse siano rilevati con fattori di scala differenti. Piuttosto che ridurre iterativamente l'immagine producendone diverse versioni a differenti livelli, risulta più conveniente aumentare le dimensioni del filtro, come mostrato in Figura 5.28. È bene sottolineare che nel caso d'utilizzo di SURF per il riconoscimento sketch - foto il fattore di scala assumerà una importanza relativa, visto che anche le immagini del volto date in input all'algoritmo sono opportunamente normalizzate in dimensioni.

Sui punti d'interesse individuati (per ulteriori dettagli sul procedimento di rilevamento si rimanda a [29]) è necessario calcolare i descrittori SURF, i quali

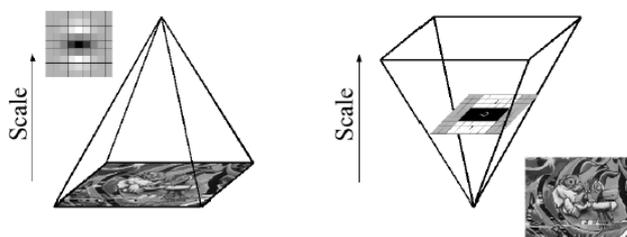


Figura 5.28: SURF: analisi a differenti valori di scala

contengono l'informazione relativa alla distribuzione dell'intensità dei pixel vicini. Piuttosto che basarsi sul gradiente, i vettori SURF sono costruiti a partire dalle risposte ai filtri di Haar del primo ordine, facendo leva sulla velocità di calcolo offerta dalle immagini integrali e utilizzando 64 dimensioni. La prima fase di costruzione dei descrittori prevede, per ogni punto d'interesse, l'individuazione dell'orientazione principale. Questa attività viene svolta calcolando la risposta ai filtri di Haar nelle direzioni  $x$  e  $y$ , all'interno di una finestra circolare centrata sul punto stesso.

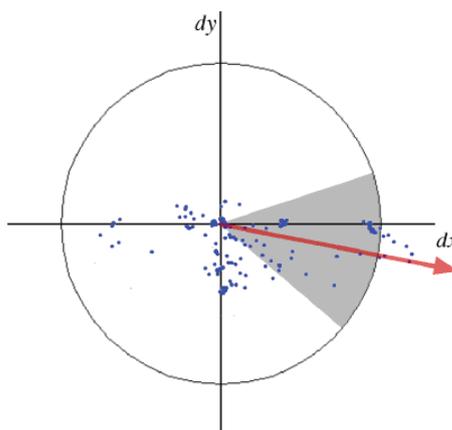


Figura 5.29: SURF: calcolo dell'orientazione dominante

Come mostra Figura 5.29, il risultato di questo processo può essere rappresentato con una serie di punti nello spazio e a partire da questi si determina l'orientazione dominante della feature. Terminata questa prima fase, si costruisce una finestra quadrata centrata sul punto d'interesse, orientata in base alla direzione dominante calcolata in precedenza e di dimensione dipendente dal fattore di scala con cui è stata individuato il punto d'interesse (Figura 5.30). La finestra quadrata costruita sul punto viene suddivisa in 16 sottoregioni quadrate di eguale dimensione, in modo da preservare le informazioni spaziali. Per ciascuna sottoregione, vengono calcolate le risposte alla funzione wavelet



Figura 5.30: SURF: esempi di finestre quadrate a dimensione e orientazione variabile centrate sui punti d'interesse individuati

di Haar; per ragioni di semplicità, indichiamo con  $d_x$  la risposta alla funzione wavelet lungo l'asse orizzontale e con  $d_y$  la risposta alla funzione wavelet lungo l'asse verticale, dove naturalmente il concetto di 'verticale' e 'orizzontale' è da intendersi in relazione alla specifica orientazione della finestra (Figura 5.31).

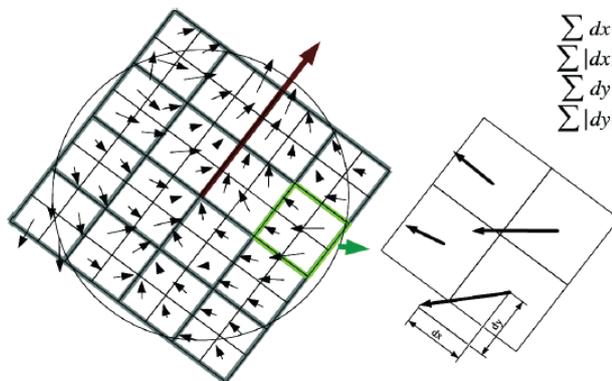


Figura 5.31: SURF: suddivisione della finestra quadrata in sottoregioni e calcolo del descrittore a 64 componenti

Per aumentare la robustezza verso deformazioni geometriche ed errori di localizzazione, la risposta alla wavelet viene pesata attraverso una Gaussiana centrata sulla zona d'interesse. A questo punto, sommando da un lato le risposte, indicate con  $dx$  e  $dy$ , ottenute da ciascuna sottofinestra e calcolando dall'altro lato la somma dei rispettivi valori assoluti, si ottiene un vettore  $\mathbf{v}$  della singola

sottofinestra caratterizzato da 4 componenti:

$$\mathbf{v} = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$$

Concatenando tutti i vettori delle 16 diverse regioni in cui è stata divisa la finestra, si ottiene un descrittore complessivo del punto d'interesse, descrittore composto appunto da 64 elementi.

L'esecuzione dell'algoritmo SURF a partire dalle immagini originali normalizzate ed allineate di sketch e foto produce un risultato come quello rappresentato in Figura 5.32.

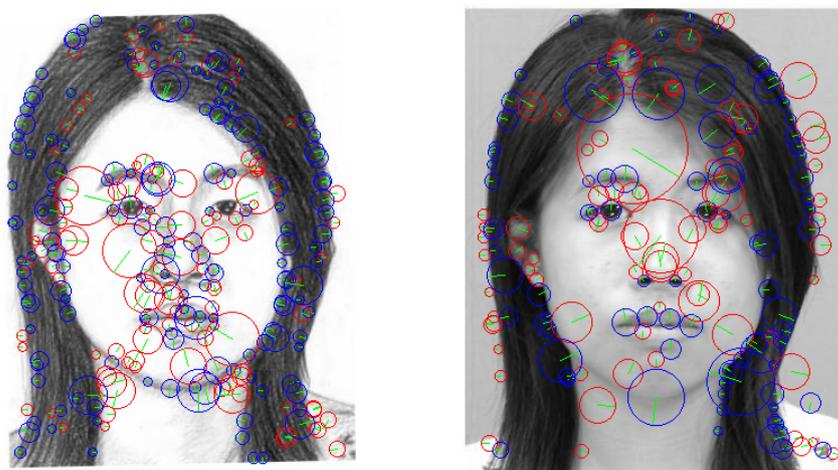


Figura 5.32: Calcolo dei punti SURF sulle immagini originali (normalizzate ed allineate) di foto e sketch

Ogni punto SURF rilevato è graficamente indicato con un circoletto ed un segmento che riassumono diverse informazioni. L'ampiezza del cerchio indica il fattore di scala con cui è stato individuato quel punto, mentre il segmento di colore verde rappresenta l'orientazione dominante della feature. Il cerchio è colorato in rosso oppure in blu a seconda che il punto rilevato sia chiaro in un background zona scura o, al contrario, scuro in un background chiaro. Questo tipo di caratteristica è espressa con un parametro chiamato Laplaciano, ed è utile per scartare a priori alcuni accoppiamenti in fase di matching. Ad ogni punto sono poi associati i 64 valori di risposta ai filtri di Haar in quella regione. La verifica di somiglianza tra una fotografia ed uno sketch sulla base dei punti rilevati consiste nel determinare i punti feature corrispondenti tra un'immagine e l'altra. Maggiore è il numero di coppie di punti tra i quali c'è una corrispondenza e maggiore è la possibilità che il soggetto rappresentato nella foto sia il medesimo di quello rappresentato nello sketch. Il matching tra i punti SURF estratti a partire dalle due immagini lavora sia a livello locale che a livello globale. Questo significa che l'algoritmo di individuazione delle corrispondenze

deve non solo assicurarsi che una coppia di feature abbia una elevata similarità per essere considerata valida, ma deve anche assicurarsi che globalmente le corrispondenze individuate siano tra loro coerenti in senso spaziale.

La prima attività che viene svolta si pone come obiettivo quello di individuare, con approccio *brute - force*, quali sono le coppie di punti sketch - foto che hanno una buona similarità e che quindi possono essere potenzialmente delle corrispondenze corrette. Pertanto, una volta selezionati due punti, di cui uno appartenente allo sketch ed uno alla foto, per valutarne localmente la similarità viene utilizzato un criterio *nearest neighbor* o di distanza euclidea che coinvolge tutte le componenti precedentemente menzionate. La distanza euclidea viene cioè valutata tra le orientazioni, i descrittori di Haar e i fattori di scala dei due punti. Combinando le similarità delle singole caratteristiche si ottiene un valore unico; le coppie che vengono considerate 'interessanti' e passano alla fase di analisi successiva sono quelle la cui similarità supera una soglia prestabilita. Alla fine di tale processo, l'algoritmo ha a disposizione una serie di coppie la cui effettiva validità va ulteriormente approfondita; entra in questo momento in gioco una tecnica nondeterministica chiamata RANSAC [50]. Pubblicato nel 1981 da Fischler e Bolles, RANSAC nasce come metodo iterativo per la stima dei parametri di un modello matematico a partire da un insieme di dati contenente outlier. È un algoritmo nondeterministico nel senso che produce un risultato corretto solo con una data probabilità, che aumenta al crescere delle iterazioni consentite. Nella fattispecie del matching tra punti, RANSAC tenta di costruire un modello che preveda il maggior numero di corrispondenze in base alla loro collocazione spaziale; i punti SURF che hanno una elevata similarità locale ma che sono spazialmente collocati in zone differenti dell'immagine non sono compatibili al fine della costruzione del modello globale e quindi vengono scartati. Al termine del procedimento, solo le coppie di punti effettivamente corrispondenti entrano a far parte del modello globale e vengono restituite come risultato.

### 5.3 Riconoscimento mediante fusione dei risultati

Nel Capitolo 3 è stato mostrato ad alto livello come il processo di riconoscimento complessivo avvenga confrontando lo sketch del soggetto da identificare con tutte le fotografie presenti in un ipotetico database. Dopo aver visto tutte le feature che sono state implementate, è venuto il momento di addentrarci più nel dettaglio della questione legata al riconoscimento. In particolare, supponiamo di dover ricercare l'identità del soggetto avendo a disposizione un database contenente  $N$  fotografie di  $N$  soggetti distinti basando il riconoscimento sull'utilizzo di una sola feature. In tal caso, occorrerebbe effettuare un solo ciclo di confronto con ognuna delle  $N$  fotografie a disposizione, ottenendo come risultato un totale di  $N$  valori di similarità calcolati sulla base della feature e

del relativo criterio di similarità impiegato per il confronto. In seguito, sarebbe sufficiente l'impiego di un banale algoritmo di ordinamento per ordinare i valori di similarità dal più alto al più basso ed ottenere una classifica delle fotografie più somiglianti, con in prima posizione la fotografia maggiormente corrispondente allo sketch. In realtà, il nostro sistema è basato su molteplici feature che possono potenzialmente essere usate assieme; questa situazione impone l'utilizzo di una tecnica di fusione dei risultati ottenuti da ogni singola feature al fine di produrre il risultato globale. In letteratura esistono molteplici tecniche di fusione e quelle che potrebbero essere a noi utili sono riassumibili nelle seguenti categorie [54]:

- fusione a livello di feature: è una tecnica che prevede di combinare assieme i descrittori ottenuti da feature diverse al fine di ottenere un nuovo vettore di feature rappresentativo dell'individuo. L'attività di fusione avviene quindi prima del matching in modo che quest'ultimo possa essere effettuato a partire dai nuovi descrittori;
- fusione a livello di score: consiste nell'utilizzare algoritmi di matching diversi che restituiscono un insieme di score che vengono poi fusi per generare un unico score finale;
- fusione a livello di rank: le diverse feature forniscono un ranking (classifica o ordinamento) delle fotografie più simili allo sketch preso come riferimento. I rank prodotti dalle feature sono poi unificati per ottenere una 'classifica finale' utile per la decisione sull'identità della persona;
- fusione a livello di decisione: ogni feature esprime una 'preferenza' su quale sia la fotografia più simile allo sketch, definendo in questo modo l'identità del soggetto. La preferenza finale viene ottenuta combinando le singole preferenze sulla base di una precisa regola, come ad esempio la maggioranza dei voti.

Potenzialmente, tutte le tecniche illustrate potrebbero essere utilizzate ma quella più interessante è sicuramente la fusione a livello di rank. La nostra volontà è infatti quella di ottenere dal sistema di riconoscimento una 'classifica' delle fotografie più simili allo sketch, favorendo la presenza del soggetto corretto nelle prime posizioni, anche se non necessariamente nella prima. Questo consente all'ipotetico utilizzatore del sistema di poter valutare un insieme di soggetti candidati, come del resto accade tipicamente nell'ambito della identificazione di un ricercato a partire dal proprio identikit. In quest'ottica, la scelta della tecnica di fusione è caduta sulla cosiddetta metodologia *borda - count*. Per capire in cosa consiste, si può utilizzare la metafora del campionato di motociclismo e dei Gran Premi. Ogni Gran Premio, infatti, vede un pilota vincitore ed altri piloti che pur non vincendo riescono ad ottenere dei punti. Alla fine della stagione, i punti ottenuti da ciascun pilota determinano la classifica finale; per aggiudicarsi il titolo di Campione, non è necessario che

un pilota abbia vinto tutte le gare ma è sufficiente che abbia complessivamente totalizzato più punti dei suoi rivali. Propagando questa metafora al nostro riconoscimento del volto, possiamo dire che il ruolo dei Gran Premi è svolto dai confronti sketch - foto sulla base della singola feature, mentre il ruolo di Campione è svolto dalla fotografia che dopo i confronti effettuati, feature dopo feature, risulta essere quella che ha totalizzato un numero maggiore di punti. In altri termini, gli  $N$  confronti tra lo sketch e le diverse foto devono essere ripetuti più volte, cambiando di volta in volta la tipologia di feature utilizzata in modo che ogni feature possa fornire una propria classifica delle foto più simili allo sketch.

L'utilizzo del sistema di fusione borda - count prevede la definizione di un sistema di classifica a punti, che permetta di associare ad ogni fotografia il suo punteggio al termine di ogni ciclo di confronti. La regola generale impone di stabilire il numero di posizioni che vanno a punti e associare di conseguenza il valore di punteggio a ciascuna posizione; in particolare, se  $n$  sono le posizioni che vanno a punti, il punteggio associato alla posizione  $p$ -esima è pari a  $n - p + 1$ . Ad esempio, se vengono premiate le prime 10 posizioni, il primo classificato totalizzerà 10 punti, il secondo 9, il terzo 8 e così via fino ad arrivare all'ultimo classificato che acquisirà un solo punto.

Nella fattispecie di questa tesi, sono state prese in esame due casistiche: un sistema di punteggio in cui vanno a punti le prime 15 fotografie e un sistema di punteggio in cui vanno a punti le prime 10. Al fine di sperimentare possibilità alternative di distribuzione dei punteggi oltre alla tecnica generale prevista da borda - count e precedente menzionata, sono stati definiti due nuovi sistemi di classifica in cui gli scarti di punteggio tra una posizione e quella successiva non sono unitari ma sono diversi a seconda dell'importanza della posizione stessa. L'idea è cioè quella di attribuire un maggiore scarto di punteggio tra le fotografie che si piazzano nei primissimi posti e quelle che invece si piazzano più indietro in classifica. Si tratta di un approccio che trae ispirazione dalle già citate corse automobilistiche e che è definito proprio per dare un peso diverso ai vari piazzamenti.

Quanto detto fino ad ora è riassunto nelle 4 colonne di Tabella 5.1; in particolare, nella colonna (b) troviamo il sistema di punteggio in cui vanno a punti le prime 15 fotografie e nel quale lo scarto tra una posizione e la successiva è unitario, mentre nella precedente colonna (a) è indicato il sistema di punteggio derivato da questo ma con distribuzione dei punti calibrata diversamente. Analogamente, nelle colonne (c) e (d) di Tabella 5.1 troviamo gli stessi concetti ripetuti per un sistema di punteggio in cui vanno a punti solamente le prime 10 fotografie. Questi sistemi sono stati implementati al fine di verificare quale potesse essere il più idoneo e i risultati ottenuti sono descritti nel Capitolo 7.

Tabella 5.1: Assegnamento dei punteggi alle fotografie ai primi 15 posti e ai primi 10 posti

Pos.	15 a punti (a)	15 a punti (b)	10 a punti (c)	10 a punti (d)
1	25	15	25	10
2	20	14	18	9
3	16	13	15	8
4	13	12	12	7
5	11	11	10	6
6	10	10	8	5
7	9	9	6	4
8	8	8	4	3
9	7	7	2	2
10	6	6	1	1
11	5	5	-	-
12	4	4	-	-
13	3	3	-	-
14	2	2	-	-
15	1	1	-	-

## Capitolo 6

# Implementazione

### 6.1 Piattaforma di sviluppo e librerie utilizzate

La descrizione del sistema di riconoscimento sketch - foto mostrata nei precedenti capitoli non deve distogliere il lettore da una questione fondamentale: tutti i passaggi e le fasi di elaborazione, dal pre - processing all'estrazione delle feature e relativa fusione dei risultati, devono essere implementate in una o più soluzioni software al fine di valutarne l'effettiva utilità. Nella fattispecie, il linguaggio di programmazione impiegato per la produzione del sistema è C# su piattaforma .NET Framework 4. Grazie all'utilizzo dell'IDE di sviluppo Visual Studio 2010 e grazie all'impiego di varie librerie dedicate all'elaborazione di immagini, è stato possibile implementare ciò che nei precedenti capitoli è stato descritto a livello teorico.

Entrando più nel dettaglio, oltre al già citato IVP Framework descritto con dovizia di particolari nel Capitolo 4, la principale libreria utilizzata è la Bio-Lab. Sviluppata nell'ambito del Biometric System Laboratory [51] presso l'Università di Bologna (sede di Cesena), essa fornisce un insieme di classi C# per svolgere operazioni sulle immagini, classificazione e riconoscimento di forme. Tali classi sono state sviluppate principalmente per scopi didattici e per semplificare lo sviluppo di algoritmi di elaborazione di immagini e pattern recognition, ma è utilizzata con successo anche per scopi di ricerca: ad esempio, il già citato framework ICAO (Capitolo 4) è basato fortemente sull'uso di Bio-Lab. Grazie alle funzionalità messe a disposizione da tale libreria, l'analisi delle immagini risulta notevolmente semplificata permettendo allo studente di concentrarsi maggiormente sugli algoritmi implementativi delle feature piuttosto che su dettagli a basso livello come l'accesso ai pixel di un'immagine o la modifica dei loro valori di intensità. Inoltre, la libreria mette a disposizione anche alcune funzioni non banali come il ridimensionamento delle immagini, il passaggio da uno spazio colore ad un altro e varie misure di distanza tra vettori che permettono di disporre di un'ottima base da cui partire per la realizzazione di funzioni più complesse.

Un'altra libreria molto importante utilizzata specificatamente nell'implementazione della feature SURF è la libreria Accord.NET [52]. In realtà, più che di una libreria, si tratta di un vero e proprio framework di lavoro: essa è un'estensione del più popolare framework AForge.NET con nuovi tool e nuove funzionalità. In particolare, grazie alle classi e ai metodi messi a disposizione da questo framework è possibile realizzare sia la fase di estrazione sia la fase di matching del descrittore illustrato nel Capitolo 5 in modo fedele a quanto previsto a livello teorico.

## 6.2 Organizzazione del prototipo software

Per testare l'efficacia delle feature di forma e di SURF per il riconoscimento sketch - foto è stata realizzata una sorta di applicazione prototipo, sfruttando le librerie e le funzionalità descritte nel precedente paragrafo. In realtà questa soluzione software non è l'unica ad essere stata prodotta visto che anche l'estrazione dei contorni del volto e relativa normalizzazione/allineamento ne richiede una. Su quest'ultima però non c'è molto altro da aggiungere rispetto a quanto già detto nel Capitolo 4, mentre è sicuramente più interessante approfondire il discorso legato al software di riconoscimento che compone in un'unica soluzione tutti i concetti illustrati nel Capitolo 5. L'applicazione prototipo lavora pertanto su immagini che sono il frutto della fase di pre - processing ed appare come in Figura 6.1.

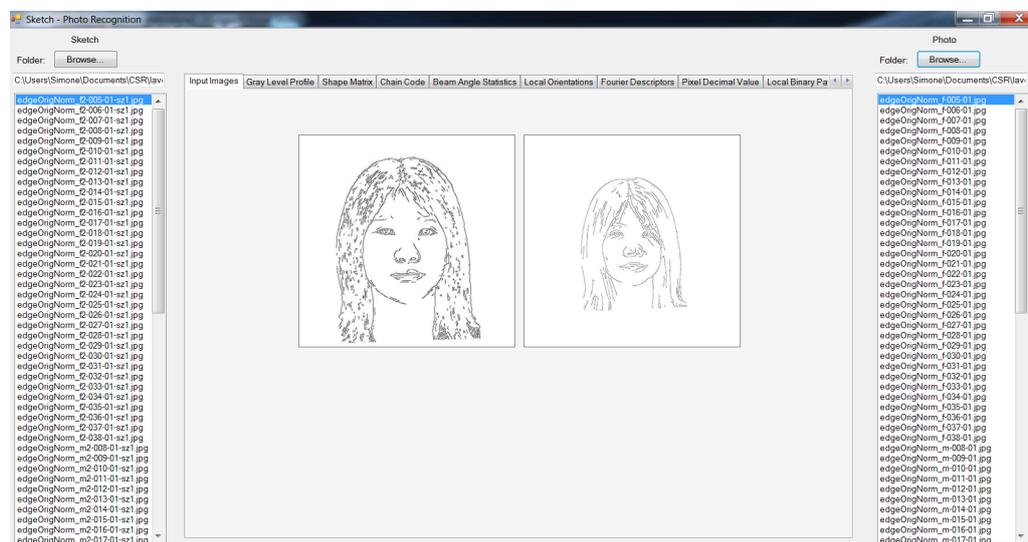


Figura 6.1: Screenshot dell'applicazione prototipo per il riconoscimento sketch - foto

Oltre a caricare le immagini dei contorni relativamente a fotografie e sketch, il

software consente di lavorare separatamente su ogni feature, andando a settare i parametri (quando previsti) e verificare gli effetti derivanti da tale impostazione. Ad esempio, in Figura 6.2 vediamo cosa accade qualora l'utente selezionasse la scheda relativa alla feature Gray Level Profile. Oltre a visualizzare i contorni di foto e sketch relativi ai volti selezionati, l'applicazione mostra quali sono gli istogrammi calcolati in base ai parametri scelti; in questo modo si può già avere una prima idea del funzionamento della feature stessa.



Figura 6.2: Screenshot dell'applicazione prototipo per la parametrizzazione di una delle feature di forma

Una volta settati i parametri, l'utilizzatore dell'applicazione può impiegare le feature stesse per il riconoscimento vero e proprio. Ad esempio, tramite la scheda mostrata in Figura 6.3 è possibile selezionare una feature per regolare il riconoscimento. Abbiamo visto nel Capitolo 5 che ogni feature ha associati uno o più criteri di misura della distanza tra descrittori: qualora per la feature scelta ne sia presente più di uno, l'utente può decidere quale utilizzare.

Selezionata la feature, è possibile calcolare la foto più simile ad un dato sketch oppure misurare la distanza sketch - foto per la relativa coppia di descrittori ed infine fare un test sull'intero dataset. Quest'ultima è la funzionalità più utilizzata in fase di testing del sistema di riconoscimento, in quanto consente di valutare le effettive prestazioni della feature selezionata. Naturalmente, come mostra Figura 6.4, è possibile ripetere le operazioni di testing globale e ricerca della foto più simile allo sketch anche selezionando più feature contemporaneamente dando vita al processo di fusione dei risultati descritto alla fine del Capitolo 5.

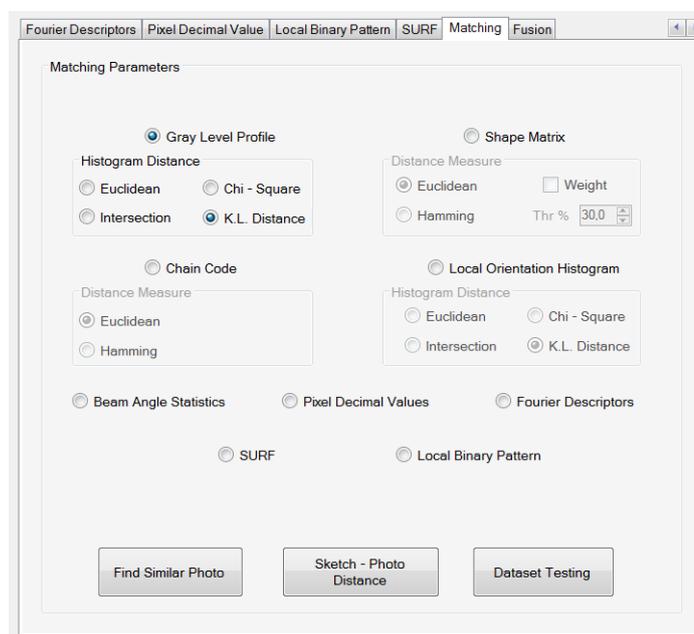


Figura 6.3: Screenshot dell'applicazione prototipo in caso d'utilizzo di una sola feature per il riconoscimento

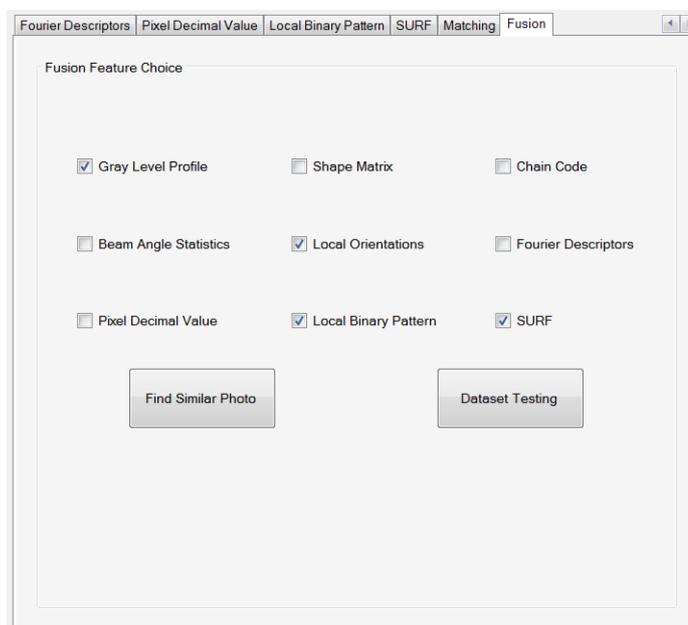


Figura 6.4: Screenshot dell'applicazione prototipo in caso d'utilizzo di più feature per il riconoscimento

## Capitolo 7

# Prove sperimentali

### 7.1 Organizzazione delle prove sperimentali

I concetti teorici illustrati e la panoramica relativa al prototipo software realizzato fanno capire come il sistema di riconoscimento sketch - foto sia particolarmente complesso e quindi non semplice da testare in tutti i suoi aspetti. Le difficoltà incontrate nello svolgimento delle prove sperimentali sono legate soprattutto alla scarsa reperibilità di immagini adatte alla tematica trattata. Il riconoscimento da sketch, oltre ad essere un argomento di ricerca piuttosto recente nell'ambito della più vasta branca del riconoscimento del volto, richiede anche la disponibilità di dataset particolari, consistenti in immagini reali dei soggetti e identikit/sketch ad essi associati. Se per quanto concerne le fotografie dei soggetti c'è ampia disponibilità di materiale grazie alla presenza sul web di svariati dataset scaricabili gratuitamente o a pagamento, la stessa cosa non si può dire dei disegni. Questi ultimi, anche quando prodotti attraverso strumenti software piuttosto che dalla mano dell'artista, richiedono un lavoro importante per far sì che essi siano quantomeno somiglianti al soggetto reale. La difficoltà oggettiva nel creare disegni relativi a centinaia di soggetti per dar vita a dataset ampi e utili per il testing rende tale 'materia prima' piuttosto difficile da reperire. Questo problema si acuisce nel caso dei veri e propri identikit: quando presenti, essi sono custoditi gelosamente negli archivi delle forze dell'ordine e risultano praticamente inaccessibili.

Per le prove sperimentali ci si è quindi affidati ai pochi dataset disponibili gratuitamente sul web, cioè CUHK (*Chinese University of Hong Kong*) e AR [53]. Il database CUHK contiene in totale 188 soggetti e per ognuno ha associati fotografia reale e disegno come mostrato in Figura 7.1. I 188 soggetti sono suddivisi in due insiemi: il primo ne contiene 88 e prende il nome *training - set* mentre il secondo contiene i restanti 100 e prende il nome di *testing - set*. L'altro database utilizzato per le prove sperimentali prende invece il nome di AR dataset e contiene in tutto immagini relative a 123 soggetti. È bene precisare che il dataset AR, per quanto concerne le fotografie reali del soggetto,

prevede molteplici immagini legate a variazioni di illuminazione e posa per uno stesso individuo. Nel contesto di questa tesi si è naturalmente considerata solo la fotografia accoppiata allo sketch di riferimento, cioè la fotografia che ritrae il soggetto in posizione frontale e in condizioni di espressione ed illuminazione standard (Figura 7.2).



Figura 7.1: Dataset CUHK: esempi di volti contenuti nel dataset



Figura 7.2: Dataset AR: esempi di volti contenuti nel dataset

Nei dataset indicati, gli sketch associati alle fotografie sono di tipo *viewed*, cioè sono prodotti da un artista che ha a disposizione la foto reale del soggetto e tenta di riprodurne fedelmente le fattezze. Gli identikit veri e propri, invece, sono ottenuti a partire dalla descrizione di un testimone e questo rende la tematica del riconoscimento più complessa rispetto all'utilizzo degli sketch *viewed*. Il lettore potrebbe quindi pensare che CUHK e AR non siano in realtà utili per effettuare prove sperimentali: a dire il vero, questi dataset sono comunque molto importanti perchè ci consentono di verificare sul campo la capacità del sistema di confrontare volti rappresentati in modalità differenti a partire dalla loro forma e dal loro aspetto. Essendo il nostro obiettivo quello di studiare le varie feature (di forma e SURF) e capire quali possano essere utili allo scopo e quali no, i dataset utilizzati sono di grande interesse anche se in essi non sono contenuti veri e propri identikit.

Le prove sperimentali effettuate si dividono in due categorie: training e testing del sistema. Il training consiste nell'effettuare varie prove al fine di determinare i parametri migliori per le varie feature e nel verificare quali siano le feature che, se combinate, consentono di ottenere i risultati migliori. Il testing consiste invece nel lanciare l'esecuzione del sistema, parametrizzato nella precedente

fase di training, su soggetti diversi da quelli utilizzati per il training verificandone in tal modo le prestazioni.

Nella varie prove sperimentali il sistema restituisce sempre una classifica delle fotografie più simili allo sketch preso in considerazione, seguendo i principi descritti nel paragrafo 5.3. Al fine di valutare la capacità del sistema di riconoscere correttamente i soggetti si è scelto di calcolare il cosiddetto *Rank - N Recognition Rate*: questo valore indica la percentuale di sketch per i quali il sistema individua la fotografia associata come ‘classificata’ in una delle prime  $N$  posizioni. È infatti d’obbligo sottolineare ancora una volta che, visto il particolare problema trattato in questa tesi, è molto importante che il soggetto cercato, qualora non venga individuato come primo classificato, compaia almeno nelle prime posizioni in modo tale da fornire all’utente del sistema un ventaglio di possibili soggetti somiglianti a quello disegnato nello sketch. Il concetto di *Rank - N Recognition Rate* consente pertanto di valutare in modo piuttosto efficace tale capacità del sistema e, per analizzarne al meglio le proprietà, si è scelto di calcolare due distinti valori *Rank - N*:

- *Rank - 1 Recognition Rate*: indica la percentuale di sketch per i quali il sistema individua come prima ‘classificata’ la foto associata;
- *Rank - 5 Recognition Rate*: indica la percentuale di sketch per i quali il sistema individua la fotografia associata come ‘classificata’ in una delle prime 5 posizioni.

Il valore di *Rank - 1* consente di valutare le prestazioni del sistema in senso assoluto, mentre considerare la percentuale di *Rank - 5* permette di capire se il sistema, in base alla configurazione adottata, sia comunque in grado di ‘avvicinarsi’ all’identità corretta. Quest’ultima percentuale è molto utile soprattutto in fase di training e parametrizzazione del sistema e permette di capire se l’esito della prova sperimentale effettuata, in presenza di una eventuale percentuale di *Rank - 1* non elevata, è da considerarsi comunque interessante oppure no. La scelta di  $N$  è caduta sul valore 5 in quanto esso rappresenta un buon compromesso e, come vedremo successivamente, i risultati sperimentali confermano la bontà di questa scelta.

## 7.2 Training del sistema

Il training del sistema, come accennato in precedenza, è stato effettuato con l’obiettivo di parametrizzare le feature e individuare le migliori combinazioni tra di esse al fine di ottenere buone prestazioni di riconoscimento. In particolare, si è agito in due fasi: in primo luogo sono state considerate separatamente le varie feature di forma e SURF per impostare i parametri migliori e solamente in un momento successivo sono stati effettuati tentativi di combinazione delle diverse feature al fine di ricercare le configurazioni più vantaggiose. In entrambi i casi gli sketch e le foto associate su cui si è lavorato sono quelle contenute

nell'insieme training - set del dataset CUHK.

Per quanto concerne la parametrizzazione delle varie feature si è proceduto principalmente per sperimentazioni successive, modificando i valori dei parametri e ripetendo le prove così da poter verificare se le modifiche conducessero o meno a incrementi nelle prestazioni di riconoscimento. Le indicazioni derivanti dalle prove sperimentali effettuate sono le seguenti:

- **Gray Level Profile:** per questa feature di forma risulta importante scegliere un numero di 'campioni' (vedi paragrafo 5.1.1) piuttosto elevato in modo da costruire istogrammi il più possibile fedeli ai contorni presenti nell'immagine. Per quanto concerne invece il fattore di raggruppamento dei bin, risulta vincente un valore piuttosto basso. Ad esempio, scegliendo un numero di 'campioni' pari a 210, il fattore di raggruppamento può essere impostato ad un valore pari a 5 producendo istogrammi finali con un numero di bin pari a 42;
- **Shape Matrix:** per quanto concerne questa feature gli unici due parametri da configurare sono il numero di righe e di colonne della matrice (vedi paragrafo 5.1.2). Al termine delle prove sperimentali i migliori risultati sono stati ottenuti impostando un valore di righe pari a 15 e di colonne pari a 10;
- **Local Orientation Histogram:** come visto nel paragrafo 5.1.3, per quanto riguarda il numero di righe e colonne della matrice da sovrapporre all'immagine del volto risulta conveniente scegliere un numero di righe pari a 6 e un numero di colonne pari a 5 in modo da avere delle celle piuttosto ampie in cui andare a calcolare le orientazioni locali e relativi istogrammi. Per quanto concerne invece il numero di intervalli con cui discretizzare lo spazio delle orientazioni, è stato scelto un valore pari a 10;
- **Fourier Descriptors:** i parametri che è necessario impostare per questa feature sono lo step size che regola l'intervallo di prelievo dei punti sul bordo esterno del volto e il numero di coefficienti di Fourier alle basse frequenze da considerare per la costruzione del descrittore finale. Il primo dei due parametri citati deve essere scelto piuttosto basso, nel nostro caso pari a 5, in modo da considerare molteplici punti del contorno esterno, mentre il numero dei descrittori è stato scelto pari a 19, un numero notevolmente ridotto rispetto alla totalità di coefficienti di Fourier effettivamente disponibili;
- **Local Binary Pattern:** come visto nel paragrafo 5.1.5, sono presenti diversi parametri associati alla feature. Per quanto concerne raggio e numero di pixel dell'intorno è conveniente optare per i valori standard, cioè un raggio pari a 1 e un 8 - intorno. Il numero di righe e colonne in cui dividere l'immagine è stato fissato pari a 11 per un totale di 121 finestre.

Infine, la soglia grayscale è pari a 250: questo significa che anche i pixel che assumono una colorazione inferiore a 255 ma superiore a 250 sono considerati comunque come background;

- **Beam Angle Statistics:** per quanto concerne questa feature sono state effettuati numerosi tentativi al fine di individuare i valori migliori per i parametri relativi al numero di punti campione da considerare sul bordo esterno del volto e al grado del cosiddetto momento  $k$  (vedi paragrafo 5.1.6). I risultati più interessanti, in particolare, si ottengono impostando la ricerca di 22 punti campione e stabilendo a 5 il valore di  $k$ ;
- **Chain Code e Pixel Decimal Value:** nel paragrafo 5.1.7 è stato mostrato come i parametri fondamentali delle due feature siano la soglia grayscale e i valori di ridimensionamento delle immagini. Per quanto concerne la soglia grayscale, essa è impostata a 250 per la feature Chain Code mentre per la feature Pixel Decimal Value è impostata a 254. A riguardo del ridimensionamento, per la feature Chain Code è previsto un ridimensionamento a grandezza  $330 \times 400$ , mentre per la feature Pixel Decimal Value si è adottato un ridimensionamento che produce un'immagine notevolmente ridotta e di dimensioni pari a  $50 \times 65$ ;
- **SURF:** la feature SURF, come visto nel paragrafo 5.2, richiede un calcolo piuttosto complesso ed è facile immaginare come i parametri coinvolti siano molteplici. Molti di questi parametri sono stati impostati secondo gli standard previsti e non sono stati fatti tentativi di calibrazione; gli unici parametri che sono stati variati riguardano il tipo di algoritmo per il matching tra punti. Come anticipato nel relativo paragrafo, si è scelto di adottare uno schema brute - force per il confronto tra i punti SURF e la tecnica RANSAC per il calcolo del modello probabilistico.

I risultati ottenuti utilizzando singolarmente le diverse feature per il riconoscimento dei soggetti del training - set CUHK (88 soggetti) sono descritti in Tabella 7.1. Come riportato in tale Tabella, per ogni feature sono presenti le due percentuali di riconoscimento di cui si è parlato nel precedente paragrafo. Le percentuali presenti fanno emergere innanzi tutto un dato importante: tutte le feature di forma prese singolarmente offrono prestazioni di riconoscimento interessanti e risultano meritevoli di essere ulteriormente approfondite mediante uso combinato. Quella che offre le prestazioni più elevate è la feature di forma Local Orientation Histogram mentre quella che risulta essere più in difficoltà nel riconoscimento è la feature Gray Level Profile. Per quanto riguarda la prima, risulta molto interessante soprattutto quel 81,8 % come Rank - 5 Recognition Rate: questo valore percentuale evidenzia la capacità della feature di individuare il soggetto corretto, se non come primo classificato, comunque nei primissimi posti e questa proprietà sarà determinante quando tale feature verrà combinata assieme alle altre. Dall'altro lato, invece, Gray Level Profile

risulta essere molto meno efficace sotto questo punto di vista. In mezzo alle due feature citate si collocano tutte le altre: ad esempio, le percentuali di riconoscimento di Beam Angle Statistics sono seconde solamente a quelle di Local Orientation Histogram ed anche le feature Pixel Decimal Value, Local Binary Pattern, Chain Code e Shape Matrix restituiscono dei risultati positivi. Interessanti risultano essere anche le prestazioni offerte dalla feature Fourier Descriptors e questo indica che effettivamente si trae beneficio nell'analizzare il volto nel cosiddetto dominio di Fourier. Per quanto riguarda infine SURF, essa è in grado di riconoscere correttamente il soggetto con efficacia superiore a quelle di qualsiasi altra feature di forma e si rivela quindi un'ottima tecnica di confronto diretto tra sketch e foto.

Terminate le prove sperimentali dedicate alla valutazione delle singole feature si è passati alle sperimentazioni per individuarne le migliori combinazioni. In particolare, queste prove sperimentali sono state condotte con un duplice obiettivo: quello di valutare le migliori combinazioni e quello di valutare il miglior sistema di punteggio. Non è da dimenticare infatti che quando si utilizzano assieme più feature si deve anche prevedere l'impiego di uno dei sistemi di punteggio descritti nel paragrafo 5.3.

Per quanto riguarda l'utilizzo combinato di più feature, si potrebbe facilmente pensare che per ottenere buone prestazioni di riconoscimento sia sufficiente considerare quelle rivelatisi migliori singolarmente. In realtà, le numerose prove effettuate hanno dimostrato che non è necessariamente così. Si prenda ad esempio in esame Tabella 7.2, la quale riporta alcuni dei migliori risultati ottenuti combinando le diverse feature tra loro; la percentuale indicata tra parentesi è la percentuale di Rank - 5 Recognition Rate, mentre l'altra è quella classica di Rank - 1 Recognition Rate. Il lettore può notare come i risultati migliori si ottengano combinando assieme le sole feature SURF, Gray Level Profile, Local Orientation Histogram, Chain Code, Beam Angle Statistics e Fourier Descriptors. In un contesto di questo tipo, la percentuale di Rank - 1 arriva a toccare l'87,5% mentre la percentuale di Rank - 5 risulta essere addirittura pari al 97,7%. In altri termini, con quelle feature e quel sistema di punteggio è altamente probabile che la fotografia associata allo sketch, qualora non compaia come prima classificata, comparirà sicuramente tra le prime 5 restituite dal sistema di riconoscimento. Le feature che invece sono risultate non efficaci in un contesto di combinazione assieme alle altre sono Pixel Decimal Value, Shape Matrix e Local Binary Pattern; Tabella 7.2 mostra proprio che in alcuni casi l'utilizzo di queste ultime feature può portare anche ad un decremento delle prestazioni.

Un altro importante aspetto che è stato necessario valutare mediante l'esecuzione di varie prove sperimentali riguarda il sistema di punteggio da adottare. Quando si utilizzano più feature in combinazione tra loro, ognuna fornisce una propria classifica delle fotografie più simili allo sketch e ad ogni fotografia è necessario associare un punteggio, come descritto nel paragrafo 5.3. I risultati ottenuti mediante l'adozione di un sistema di punteggio in cui vanno a punti le

Tabella 7.1: Risultati ottenuti sul training - set CUHK (88 soggetti) utilizzando singolarmente le diverse feature

<b>Nome Feature</b>	<b>Rank - 1 RR</b>	<b>Rank - 5 RR</b>
Gray Level Profile	27,2%	43,1%
Shape Matrix	40,9%	68,1%
Local Orientation Hist.	61,3%	81,8%
Fourier Descriptors	43,1%	60,2%
Local Binary Pattern	47,7%	60,2%
Beam Angle Statistics	51,1%	70,4%
Chain Code	31,8%	52,2%
Pixel Decimal Value	46,5%	67,0%
SURF	71,5%	88,6%

Tabella 7.2: Alcune combinazioni di feature che consentono di ottenere i migliori risultati in termini di accuratezza di riconoscimento sul dataset di training; la percentuale indicata tra parentesi è la percentuale di Rank - 5 Recognition Rate, mentre l'altra è quella di Rank - 1 Recognition Rate

<b>GLP</b>	<b>SM</b>	<b>CC</b>	<b>LOH</b>	<b>BAS</b>	<b>FD</b>	<b>PDV</b>	<b>LBP</b>	<b>SURF</b>	<b>%</b>
x	x	x	x	x	x	-	-	x	80,6 (98,8)
x	-	x	x	x	x	x	-	x	80,6 (97,7)
x	-	x	x	x	x	-	x	x	81,8 (97,7)
x	-	x	x	x	x	-	-	x	87,5 (97,7)
x	-	-	x	x	x	x	-	x	82,9 (97,7)
x	-	-	x	x	x	-	x	x	82,9 (98,8)
x	-	-	x	x	x	-	-	x	82,9 (98,8)

prime 15 fotografie ed in cui vanno a punti solamente le prime 10 sono piuttosto simili, ma quest'ultimo è quello che ha consentito di ottenere le percentuali di riconoscimento migliori. Inoltre, la tecnica di distribuzione dei punti che si è rivelata maggiormente efficace è quella che prevede di distribuire i punti con scarto non unitario tra una posizione e quella successiva. Per capire meglio di cosa stiamo parlando, consideriamo Figura 7.3; essa riporta le percentuali di Rank -  $N$  Recognition Rate con  $N$  che varia tra 1 e 10 e con l'adozione di due distinte tecniche di distribuzione dei punti.

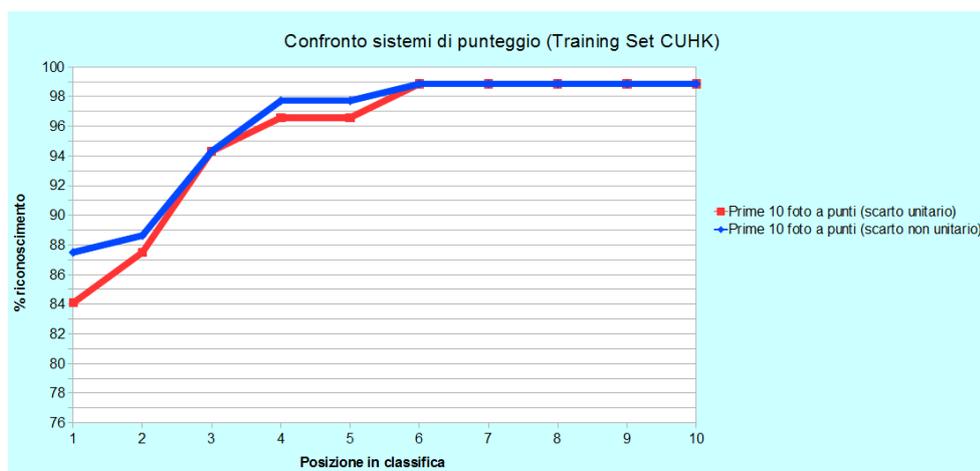


Figura 7.3: Training Set CUHK: confronto tra le percentuali di Rank -  $N$  Recognition Rate in caso d'adozione di un sistema di punteggio a scarto unitario e non unitario

In particolare, la linea rossa riguarda le prestazioni ottenute con un sistema di punteggio in cui vanno a punti le prime 10 fotografie e con distribuzione dei punti a scarto unitario, mentre la linea blu riguarda la medesima situazione con distribuzione dei punti a scarto non unitario (vedi Tabella 5.1 (c) e (d)). Da questo grafico si desume che l'adozione di un sistema di punteggio in cui i punti vengono distribuiti dando un maggiore peso alle prime posizioni rispetto alle ultime e premiando in modo diverso i vari 'piazziamenti' consente di ottenere risultati migliori rispetto ad un sistema di punteggio in cui i punti si decrementano di una unità di posizione in posizione. I risultati riportati nella precedente Tabella 7.2 sono stati ottenuti proprio considerando questo sistema di punteggio.

Un'altra importante indicazione che ci viene data dal grafico riportato in Figura 7.3 riguarda la capacità del sistema di localizzare la fotografia associata allo sketch nelle primissime posizioni della classifica finale; si può infatti notare che, indipendentemente dal sistema di punteggio adottato, è già sufficiente considerare le fotografie classificate nelle prime 5 posizioni per sperare di trovare fra di esse quella correttamente associata allo sketch visto che la probabilità di

riconoscimento tocca in questo caso quasi il 97%. Questo conferma anche che, nella valutazione delle prestazioni delle singole feature, quella di stabilire  $N = 5$  è stata una scelta corretta.

### 7.3 Testing del sistema

La fase di training descritta nel precedente paragrafo ha consentito di parametrizzare il sistema e di individuare una combinazione di feature in grado di ottenere ottime prestazioni di riconoscimento. A questo punto, occorre considerare il sistema nella miglior configurazione possibile emersa durante il training per verificare se tale ‘calibrazione’ offra buoni risultati anche su nuovi soggetti diversi da quelli considerati fino a questo momento. Al fine di testare ulteriormente l’effettiva capacità del sistema di riconoscere il soggetto a partire dal proprio sketch si rende necessario ripetere nuove prove sperimentali su dataset diversi da quello utilizzato durante il training. In particolare, le nuove prove sperimentali sono state condotte sull’insieme testing del database CUHK e su AR dataset: come descritto nel primo paragrafo di questo capitolo, il primo contiene coppie foto - sketch relative a 100 soggetti diversi, mentre il secondo ne contiene 123.

Come il lettore può notare, i nuovi dataset possiedono un numero di individui superiore rispetto al dataset utilizzato per il training; ciò che ci si attende è quindi un decremento delle prestazioni rispetto al caso ottimo individuato in fase di training visto che aumentano le possibilità di avere soggetti somiglianti. Questo tipo di scenario si verifica ad esempio lanciando in esecuzione il sistema sul testing - set CUHK: considerando la combinazione di feature rivelatasi migliore durante il training (SURF, Gray Level Profile, Local Orientation Histogram, Chain Code, Beam Angle Statistics e Fourier Descriptors) e adottando il sistema di punteggio in cui vanno a punti le prime 10 fotografie con scarto non unitario, la percentuale di Rank - 1 ottenuta è pari al 77% mentre la percentuale di Rank - 5 è del 94%. Come si nota, la percentuale di Rank - 1 scende del 10% circa rispetto a quella migliore ottenuta in fase di training, mentre la percentuale di Rank - 5 si mantiene piuttosto alta: ciò significa che diminuisce la precisione assoluta del sistema ma esso mantiene una buonissima capacità di individuare correttamente il soggetto almeno tra le prime 5 posizioni. Svolgendo ulteriori prove sperimentali su questo dataset e tentando nuove combinazioni di feature si è notato che considerando le sole feature Beam Angle Statistics, Fourier Descriptors, Local Orientation Histogram e SURF le prestazioni migliorano: la percentuale di Rank - 1 passa dal 77% all’82% mentre quella di Rank - 5 arriva al 96%. In altri termini, non considerando le feature di forma Gray Level Profile e Chain Code il sistema incrementa la propria capacità di riconoscimento. Per quanto concerne invece il sistema di distribuzione dei punti, le prove sperimentali effettuate confermano il giovamento che si trae dando un maggiore peso alle prime posizioni rispetto

alle ultime e premiando in modo diverso i vari ‘piazamenti’. A tal proposito, in Figura 7.4 possiamo visualizzare un grafico del tutto simile a quello di Figura 7.3; esso dimostra come considerando questo sistema di distribuzione dei punti si ottengano migliori risultati e conferma la capacità del sistema di individuare correttamente la foto associata allo sketch nelle primissime posizioni della classifica finale.

Ripetendo le medesime prove sperimentali sul dataset AR il risultato è piuttosto sorprendente: considerando sempre la migliore combinazione di feature individuata in fase di training con abbinato il medesimo sistema di punteggio, la percentuale di Rank - 1 risulta essere del 84,5% mentre quella di Rank - 5 del 94,3%. Ciò significa che, nonostante il cambio di tipologia di dataset e nonostante l’aumento notevole di individui (da 88 a 123), le prestazioni si mantengono su livelli piuttosto alti rispetto a quelle migliori ottenute durante il training, soprattutto per quanto concerne la precisione assoluta del sistema. Anche per quanto riguarda il dataset AR e le prove sperimentali eseguite su di esso è possibile visualizzare in Figura 7.5 il grafico relativo al confronto tra i due diversi metodi di distribuzione dei punti, sempre tenendo conto che il sistema di punteggio migliore è ancora una volta quello che prevede di mandare a punti le prime 10 fotografie classificate. L’andamento delle percentuali di riconoscimento al variare delle prime  $N$  posizioni considerate conferma quanto visto nei casi precedenti; in questo caso i benefici derivanti dall’utilizzo di un sistema di distribuzione dei punti con scarto non unitario sono meno consistenti ma comunque presenti. Inoltre, le prove sperimentali eseguite sul dataset AR e riassunte in questo grafico confermano ancora una volta la buona capacità del sistema di riconoscere di localizzare correttamente la fotografia associata allo sketch nelle prime posizioni in classifica.

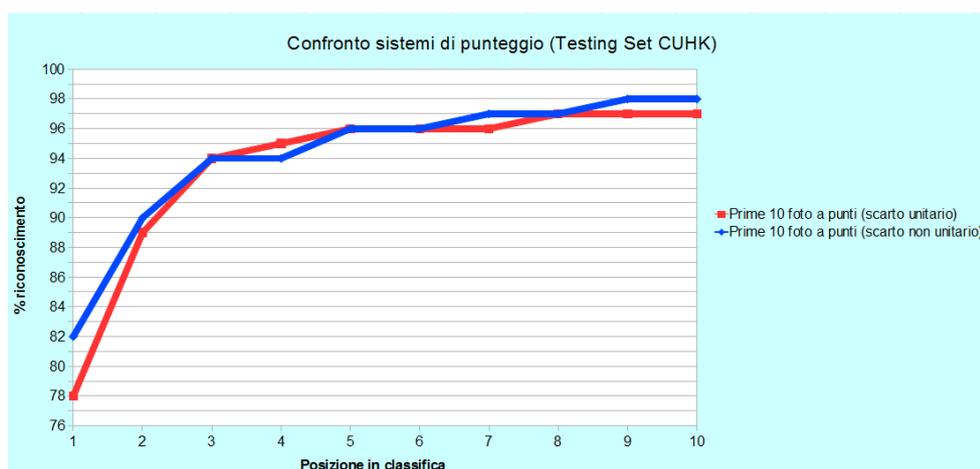


Figura 7.4: Testing Set CUHK: confronto tra le percentuali di Rank - 5 Recognition Rate in caso d’adozione di un sistema di punteggio a scarto unitario e non unitario

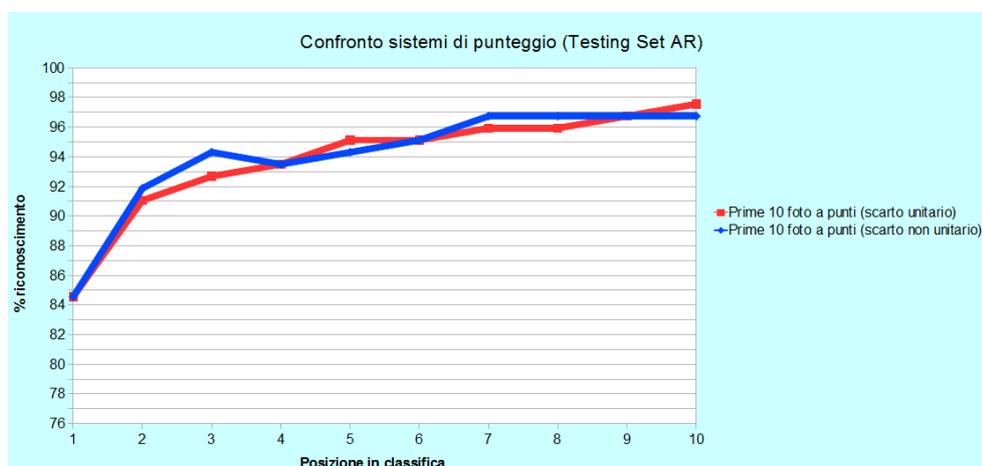


Figura 7.5: Testing Set AR: confronto tra le percentuali di Rank -  $N$  Recognition Rate in caso d'adozione di un sistema di punteggio a scarto unitario e non unitario

Come il lettore avrà notato nella lettura di questo capitolo, sia per quanto concerne il training del sistema sia per quanto riguarda il testing, le prestazioni che sono state valutate con maggiore attenzione sono quelle riguardanti l'effettiva capacità del sistema di riconoscere correttamente i soggetti. In realtà, c'è un ulteriore aspetto di cui va tenuto conto in fase di valutazione complessiva del sistema e riguarda la sua efficienza computazionale. Risulta infatti evidente che un sistema capace di associare correttamente la fotografia al proprio sketch deve anche essere in grado di effettuare questo compito in un lasso di tempo relativamente breve. Nel caso specifico del sistema studiato e realizzato in questa tesi, risulta evidente che il tempo impiegato per l'elaborazione dipende fortemente da due fattori: il numero di feature coinvolte nel riconoscimento (maggiori sono le feature coinvolte e maggiore è il tempo impiegato dal sistema per restituire un risultato) e la dimensione e quantità delle immagini da processare. Per quanto riguarda le feature, c'è una notevole disparità a livello di complessità computazionale tra le feature di forma e SURF. Le prime infatti non richiedono elaborazioni particolarmente complesse e i relativi descrittori possono essere calcolati in tempi piuttosto contenuti. SURF, invece, soprattutto a causa dell'elevata complessità dell'algoritmo che ne sta alla base, richiede un tempo di elaborazione non trascurabile. Queste tempistiche vengono ulteriormente dilatate qualora si faccia riferimento a dataset di immagini piuttosto ampi. Nel nostro caso, comunque, il sistema ha dimostrato di avere tempi di esecuzione accettabili anche grazie all'adozione di appositi accorgimenti come l'utilizzo, quando possibile, di librerie standard ottimizzate.

## Capitolo 8

# Conclusioni e sviluppi futuri

### 8.1 Riepilogo dello studio effettuato

L'obiettivo di questa tesi è stato quello di studiare tecniche innovative per il riconoscimento del volto da identikit. Questo argomento, pur essendo collocabile dal punto di vista scientifico all'interno del più ampio ambito del riconoscimento biometrico del volto, presenta alcune importanti peculiarità che lo rendono particolarmente interessante ed innovativo. In particolare, il principale problema da affrontare consiste nella differente modalità di rappresentazione (gap di modalità) del volto; nel riconoscimento da sketch/identikit l'identificazione del soggetto avviene attraverso una serie di confronti tra un disegno del volto del soggetto chiamato sketch (prodotto da un artista o da uno strumento software) e una serie di immagini fotografiche reali. Per il confronto tra lo sketch e l'immagine reale si possono seguire diversi approcci e quello studiato in questa tesi è di tipo discriminativo: il confronto sketch - foto deve avvenire direttamente senza effettuare conversioni preliminari di una modalità in un'altra o viceversa. Nello specifico, la metodologia studiata prevede innanzi tutto una fase di pre-processing che consiste principalmente nell'estrazione dei contorni sia sulle fotografie reali del volto, sia sugli sketch. L'estrazione dei contorni, accompagnata da una attività di normalizzazione geometrica, consente di aggirare l'ostacolo del cosiddetto gap di modalità permettendo di mettere sullo stesso piano concettuale le raffigurazioni dei volti ottenuti da una macchina fotografica con quelle ottenute dal lavoro di un disegnatore.

Per valutare la somiglianza tra una foto ed uno sketch a partire dalle rispettive immagini contenenti i soli contorni del volto, si è deciso di utilizzare diverse tipologie di feature. La maggior parte delle feature impiegate prendono il nome di feature di forma; esse nascono in letteratura con lo scopo di fornire dei descrittori utili a distinguere forme diverse tra loro e trovano larga applicazione nei sistemi di image - retrieval. Nel contesto di questa tesi si è presa in prestito questa idea e si è deciso di applicarla al riconoscimento sketch - foto. In altri termini, ogni volto è per noi una forma diversa e le feature di forma

devono aiutare a discriminare il volto di un soggetto rispetto a quello di un altro, permettendo di capire se il volto disegnato in un identikit sia presente all'interno di un ipotetico database contenente fotografie reali di svariati soggetti. Le feature di forma utilizzate sono in tutto 8: Gray Level Profile, Shape Matrix, Local Orientation Histogram, Fourier Descriptors, Local Binary Pattern, Beam Angle Statistics, Chain Code e Pixel Decimal Value. Ogni feature di forma ha un proprio algoritmo di calcolo che, se eseguito sull'immagine dei contorni del volto, permette di ottenere un descrittore che riassume in forma numerica determinate peculiarità legate all'aspetto del viso.

Alle feature di forma si è aggiunto lo studio e l'impiego di un ulteriore descrittore chiamato SURF (Speeded - Up Robust Features). A differenza delle feature di forma, SURF lavora direttamente sulle fotografie e sugli sketch originali e non sui loro contorni. L'algoritmo alla base di questo descrittore prevede di individuare ed analizzare alcuni punti interessanti all'interno dell'immagine e di fornire per ognuno di essi una descrizione numerica.

Al fine di valutare le prestazioni di queste feature per il riconoscimento sketch - foto, esse sono state tutte implementate (in linguaggio C#) dando vita ad un prototipo software in grado di restituire una 'classifica' delle fotografie più simili ad un dato sketch.

## 8.2 Risultati ottenuti

Grazie al prototipo software implementato è stato possibile eseguire numerose prove sperimentali al fine di valutare a livello pratico la metodologia studiata. A causa della scarsa reperibilità di veri e propri identikit con relative fotografie associate, si è fatto riferimento a dataset presenti gratuitamente sul web che contengono delle accoppiate foto - sketch in cui lo sketch è disegnato appositamente da un artista che prende spunto direttamente dal soggetto reale. In particolare, i dataset utilizzati sono CUHK, contenente in totale 188 soggetti, e AR composto da 123 soggetti.

I dataset disponibili sono stati impiegati in parte per l'addestramento o training del sistema ed in parte per il testing. Il training è consistito in una parametrizzazione del sistema e nella ricerca della migliore combinazione di feature per il riconoscimento, mentre il testing è stato effettuato su soggetti diversi da quelli utilizzati nel training con lo scopo di verificare l'effettiva validità del sistema di riconoscimento.

I risultati ottenuti dimostrano che l'uso combinato delle feature Gray Level Profile, Local Orientation Histogram, Fourier Descriptors, Beam Angle Statistics, Chain Code e SURF assieme ad un opportuno sistema di fusione dei risultati permette di dar vita ad un sistema in grado di confrontare correttamente sketch e fotografie. In particolare, il sistema realizzato è in grado di restituire come risultato una 'classifica' delle fotografie più simili ad un dato sketch e le prove sperimentali hanno dimostrato che la fotografia associata

allo sketch compare nelle primissime posizioni della classifica con probabilità nettamente superiori al 90%.

### 8.3 Possibili sviluppi futuri

Lo studio che è stato effettuato e descritto in questa tesi rappresenta solamente un primo passo verso la creazione di un sistema di riconoscimento sketch - foto che sia il più possibile efficace ed efficiente; il sistema prototipo che è stato implementato, oltre a poter essere utilizzato così com'è, può essere oggetto di ulteriori approfondimenti e sviluppi futuri.

La sfida più complessa che può essere affrontata nel futuro consiste nel rendere questo sistema il più possibile invariante rispetto alla tipologia di sketch/identikit a disposizione. Nella realizzazione dei disegni del volto non è al momento previsto alcuno standard e questo rende oggettivamente molto complesso lo studio e lo sviluppo. Inoltre, questa sfida è resa ancora più ardua dalla scarsa reperibilità dei veri e propri identikit, motivo per il quale si è spesso costretti a fare affidamento su disegni di tipo diverso.

A livello pratico, per raggiungere questi obiettivi, si può innanzi tutto provare a rielaborare le feature già implementate. Alcune delle feature rivelatisi meno performanti possono essere modificate in modo da incrementarne l'efficacia. Inoltre, possono essere studiate nuove metodologie di confronto tra descrittori oltre che tecniche aggiuntive di fusione dei risultati ottenuti dall'impiego combinato di più feature.

Un ulteriore sviluppo futuro potrebbe consistere nell'aggiungere nuove feature oltre a quelle già presenti. In particolare, esistono numerose altre feature di forma che possono essere impiegate per il riconoscimento da sketch. Non va dimenticato infatti che le feature prese in esame in questa tesi sono state selezionate da contesti di ricerca non legati al riconoscimento del volto e quindi nulla vieta di poter individuare altre feature di forma in grado di fare lo stesso percorso logico.

Oltre a nuove feature che lavorino sui contorni degli sketch e delle fotografie, ulteriori sviluppi futuri possono riguardare lo studio di estrattori in grado di operare direttamente sulle immagini originali. Nel contesto di questa tesi SURF ha dimostrato che questo è possibile e nulla vieta di poter lavorare su nuove tecniche ed algoritmi da aggiungersi a quelli già presenti.

Senza ombra di dubbio, la tematica del riconoscimento da identikit rappresenta una delle sfide più intriganti e interessanti nell'ambito del riconoscimento biometrico e il sistema realizzato può essere visto come un punto di partenza.

# Bibliografia

- [1] J. Benson and D. I. Perrett, "Perception and recognition of photographic quality facial caricatures: implications for the recognition of natural images", *European Journal of Cognitive Psychology*, 1991, pp. 105-135.
- [2] V. Bruce, E. Hanna, N. Dench, P. Healy, and A. M. Burton, "The importance of mass in line drawings of faces. Applied Cognitive Psychology", *Applied Cognitive Psychology*, 1992, pp. 619-628.
- [3] V. Bruce and G. W. Humphreys, "Recognizing objects and faces", *Visual Cognition*, 1994, pp. 141-180.
- [4] G. M. Davies, H. D. Ellis, and J. W. Shepherd, "Face recognition accuracy as a function of mode of representation", *Journal of Applied Psychology*, 1978, pp. 180-187.
- [5] G. Rhodes and T. Tremewan, "Understanding face recognition: caricature effects, inversion, and the homogeneity problem", *Visual Cognition*, 1994, pp. 275-311.
- [6] H.S. Bhatt, S. Bharadwaj, R. Singh, M. Vatsa, "Memetically Optimized MCWLD for Matching Sketches with Digital Face Images", *IEEE Transactions on Information Forensics and Security*, 2011.
- [7] X. Tang and X. Wang, "Face photo recognition using sketch", *Proceedings of International Conference on Image Processing*, 2002, vol. 1, pp. 257-260.
- [8] X. Tang and X. Wang, "Face sketch synthesis and recognition", *Proceedings of International Conference on Computer Vision*, 2003, vol. 1, pp. 687-694.
- [9] Q. Liu, X. Tang, H. Jin, H. Lu and S. Ma, "A nonlinear approach for face sketch synthesis and recognition", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, 2005, vol. 1, pp. 1005-1010.

- 
- [10] Y. Sun, Z. Miao, Y. Wang and Z. Wang, "Automatic personalized facial sketch based on ASM", *International Conference on Image Analysis and Signal Processing*, 2010.
- [11] C. Tu and J. J. Lien, "Automatic location of facial feature points and synthesis of facial sketches using direct combined model", *IEEE Transactions on Systems, Man and Cybernetics*, 2010, pp. 1158-1169.
- [12] Y. Li, M. Savvides and V. Bhagavatula, "Illumination Tolerant Face Recognition Using a Novel Face From Sketch Synthesis Approach and Advanced Correlation Filters", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2006, vol. 2.
- [13] X. Wang and X. Tang, "Face photo - sketch synthesis and recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009, vol. 31, no. 11, pp. 1955-1967.
- [14] R. Uhl and N. Lobo, "A framework for recognizing a facial image from a police sketch", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, 1996, pp. 586-593.
- [15] P. Yuen and C. Man, "Human face image searching system using sketches", *IEEE Transactions on Systems, Man and Cybernetics - A*, 2007, vol. 37, no. 4, pp. 493-504.
- [16] Y. Zhang, C. McCullough, J. Sullins and C. Ross, "Human and computer evaluations of face sketches with implications for forensic investigations", *Proceedings of the International Conference on Biometrics: Theory, Applications and Systems*, 2008, pp. 1-7.
- [17] Y. Zhang, C. McCullough, J. Sullins and C. Ross, "Hand-drawn face sketch recognition by humans and a PCA-based algorithm for forensic applications", *IEEE Transactions on Systems, Man and Cybernetics - A*, 2010, vol. 40, no. 3, pp. 475 - 485.
- [18] H. Nizami, Adkins - Hill, P. Jeremy, Y. Zhang, J. Sullins, X. McCullough, S. Canavan and L. Yin, "A biometric database with rotating head videos and hand - drawn face sketches", *Proceedings of International Conference on Biometrics: Theory, Applications and Systems*, 2009, pp. 38 - 43.
- [19] B. Klare and A. Jain, "Sketch-to-photo matching: a feature-based approach", *Proceedings of Society of Photo-Optical Instrumentation Engineers Conference Series*, 2010, vol. 7667.
- [20] B. Klare, L. Zhifeng and A. Jain, "Matching forensic sketches to mug shot photos", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, vol. 33, pp. 639-646.

- 
- [21] B. Klare and A. Jain, "Heterogeneous face recognition using kernel prototype similarities", Tech. Rep., Michigan State University, 2011.
- [22] H. Bhatt, S. Bharadwaj, R. Singh, and M. Vatsa, "On matching sketches with digital face images", *Proceedings of International Conference on Biometrics: Theory Applications and Systems*, 2010.
- [23] W. Zhang, X. Wang and X. Tang, "Coupled information-theoretic encoding for face photo-sketch recognition", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, 2011.
- [24] A. Del Bimbo and P. Pala, "Visual Image Retrieval by Elastic Matching of User Sketches", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997.
- [25] T. Hisamori and G. Ohashi, "Query-by-Sketch Interactive Image Retrieval Using Rough Sets", *IEEE Transactions on Systems, Man and Cybernetics Society*, 2007.
- [26] K. Hildebrand, T. Boubekeur and M. Alexa, "Sketch-Based Image Retrieval: Benchmark and Bag-of-Features Descriptors", *IEEE Transactions on Visualization and Computer Graphics*, 2011.
- [27] M. Eitz, K. Hildebrand, T. Boubekeur and M. Alexa, "A descriptor for large scale image retrieval based on sketched feature lines", *Proceedings of the 6th Eurographics Symposium on Sketch-Based Interfaces and Modeling*, 2009.
- [28] M. Yang, K. Kpalma and J. Ronsin, "A survey of shape feature extraction techniques", *Pattern Recognition*, 2008.
- [29] H. Bay, A. Ess, T. Tuytelaars and L. Van Gool, "Speeded - Up Robust Features (SURF)", 2008.
- [30] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*, 2004, pp. 60(2):91-110.
- [31] J. Canny, "A Computational Approach to Edge Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679 - 698, Nov. 1986.
- [32] J. Klima, "System for Image Vectorization", 2007.
- [33] J. Klima and T. Skopal, "Shape Extraction Framework for Similarity Search in Image Databases", 2007.
- [34] S. V. Ablameyko, "Introduction to Interpretation of Graphic Images", *SPIE*, 1997.

- [35] P. Altman, "Digitalization of map", *Master's thesis, Charles University, Department of Software and Computer Science Education*, 2004.
- [36] M. Ferrara, A. Franco, D. Maio and D. Maltoni, "Face Image Conformance to ISO/ICAO Standards in Machine Readable Travel Documents", *IEEE Transactions on Information Forensics and Security*, Vol. 7, No. 4, 2012.
- [37] D. Maio and D. Maltoni, "Real-time face location on gray-scale static images", *Pattern Recognition*, vol. 33, no. 9, pp. 1525-1539, 2000.
- [38] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 511-518, 2001.
- [39] M. Dobes et al., "Human eye localization using the modified Hough transform", *Optik-Int. J. Light Electron Opt.*, vol. 117, no. 10, pp. 468-473, 2006.
- [40] A. L. Yuille, P.W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable templates", *Int. J. Comput. Vis.*, vol. 8, no. 2, pp. 99-111, 1992.
- [41] [http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL\\_COPIES/WAHL1/node5.html](http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/WAHL1/node5.html)
- [42] J. M. Saavedra and B. Bustos, "An Improved Histogram of Edge Local Orientations for Sketch-Based Image Retrieval", *German Association for Pattern Recognition DAGM*, 2010.
- [43] D. Maltoni, D. Maio, A. K. Jain and S. Prabhakar, "Handbook of Fingerprint Recognition".
- [44] [http://bias.csr.unibo.it/franco/SB/DispensePDF/3\\_Impronte\%20digitali\\_feature\%20extraction.pdf](http://bias.csr.unibo.it/franco/SB/DispensePDF/3_Impronte\%20digitali_feature\%20extraction.pdf)
- [45] D. Zhang and G. Lu, "A comparative study of fourier descriptors for shape representation and retrieval", *Proc. 5th Asian Conference on Computer Vision*, 2002.
- [46] D. Zhang and G. Lu, "A comparative study of curvature scale space and fourier descriptors for shape-based image retrieval", *Visual Communication and Image Representation*, vol. 14(1), 2003.
- [47] T. Ojala, M. Pietikainen and D. Harwood, "A comparative study of texture measures with classification based on feature distributions", *Pattern Recognition*, vol. 29, no. 1, pp. 51-59, 1996.

- 
- [48] N. Arica and F. T. Y. Vural, "BAS: a perceptual shape descriptor based on the beam angle statistics", *Pattern Recognition Letters*, no. 9-10, 1627-1639, 2003.
- [49] Y. K. Liu, W. Wei, P. J. Wang and B. Zalik, "Compressed vertex chain codes" *Pattern Recognition*, vol. 40(11), pp. 2908-2913, 2007.
- [50] <http://en.wikipedia.org/wiki/RANSAC>
- [51] <http://biolab.csr.unibo.it/home.asp>
- [52] <http://code.google.com/p/accord/>
- [53] <http://mmlab.ie.cuhk.edu.hk/cufsf/>
- [54] [http://bias.csr.unibo.it/franco/SB/DispensePDF/13\\_Sistemi%20multibiometrici.pdf](http://bias.csr.unibo.it/franco/SB/DispensePDF/13_Sistemi%20multibiometrici.pdf)

# Ringraziamenti

Giunto al termine di questo lavoro desidero ringraziare ed esprimere la mia riconoscenza nei confronti di tutte le persone che, in modi diversi, mi sono state vicine e hanno permesso e incoraggiato sia i miei studi che la realizzazione e stesura di questa tesi.

Innanzitutto, vorrei ringraziare il relatore Prof.ssa Annalisa Franco per la fiducia fin da subito dimostratami e per avermi seguito durante lo svolgimento del lavoro con consigli e confronti che mi hanno aiutato ad intraprendere, ogni volta, le scelte più appropriate. La ringrazio per la disponibilità concessami nel fornirmi chiarimenti, per la rilettura critica di tutti i capitoli della tesi e per avermi guidato con i suoi suggerimenti durante la conclusione di questo percorso formativo. Il tutto nonostante i numerosi impegni e, soprattutto, la mia pedanteria e insistenza.

Per quanto riguarda invece l'intero percorso scolastico (universitario e pre-universitario) fin qui svolto e di cui questa tesi rappresenta un importante traguardo, vorrei ringraziare la mia famiglia. Ringrazio mio padre Claudio e mia madre Mara che non mi hanno mai fatto mancare nulla; grazie al loro aiuto ho potuto percepire le realtà difficili della vita di questi tempi come qualcosa di lontano.

Infine, non posso non salutare tutti i compagni di corso con cui ho trascorso 2 anni e mezzo di lezioni ed esami continui. Con loro ho condiviso appunti, lunghe discussioni su svariati temi, timori per gli esami, preoccupazioni per la realizzazione dei famigerati progetti e soprattutto tanti momenti di divertimento.

Grazie a tutte queste persone il tempo trascorso all'università è stato, per gran parte, un immenso piacere !