



UNIVERSITÀ DEGLI STUDI DI PADOVA

FACOLTÀ DI INGEGNERIA

Corso di Laurea Specialistica in Ingegneria Informatica

BIOMETRICHE DELL'ORECCHIO
METODI DI REGISTRAZIONE E RICONOSCIMENTO

Laureando

Marco Cassol

Relatore

Prof. Carlo Ferrari

ANNO ACCADEMICO 2012/2013

Alla mia famiglia.

Indice

1	Registrazione 2D. A novel ray analogy for enrolment of ear biometrics	5
1.1	Leggi dell'ottica	6
1.1.1	Indice di rifrazione	6
1.1.2	Legge della rifrazione	7
1.1.3	Riflessione totale	8
1.2	Filtri	8
1.2.1	Filtri spaziali	8
1.2.2	Filtro del gradiente	14
1.2.3	Approssimazione del filtro gradiente	16
1.2.4	Filtro di Sobel	18
1.3	Algoritmo proposto	20
1.4	Parametri	23
1.5	Risultati	25
2	Registrazione 3D. An efficient technique for ear detection in 3D: Invariant to rotation and scale	27
2.1	Filtri	27
2.1.1	Operatore derivata della Gaussiana (DroG)	27
2.1.2	Operatore di Canny	28
2.2	Pre-processamento	32
2.2.1	Calcolo dell'immagine di profondità	32
2.2.2	Calcolo dei bordi	32
2.2.3	Approssimazione dei bordi	33
2.2.4	Eliminazione dei bordi irrilevanti	33
2.3	Calcolo dell'insieme candidato dell'orecchio	33
2.3.1	Costruzione del grafo di connettività dei bordi	34
2.3.2	Calcolo delle componenti connesse	34
2.3.3	Calcolo degli orecchi candidati	34
2.4	Localizzazione dell'orecchio	35
2.4.1	Creazione del modello d'orecchio	35

2.4.2	Identificazione dell'orecchio	38
2.5	Scala e rotazione invarianza	38
2.5.1	Invarianza alla scala	38
2.5.2	Invarianza alla rotazione	38
2.6	Risultati	39
2.6.1	Valutazione delle prestazioni	39
3	Riconoscimento 2D. An efficient ear recognition technique invariant to illumination and pose	41
3.1	Prerequisiti	42
3.1.1	Operatore gradiente direzionale	42
3.1.2	Operatore laplaciano	42
3.1.3	Haar wavelet	45
3.1.4	Trasformata Speeded Up Robust Feature (SURF) . . .	46
3.1.5	Equalizzazione adattiva a istogramma	47
3.1.6	Filtro a mezzi non locali	48
3.1.7	Filtro orientabile	49
3.2	Algoritmo proposto	50
3.2.1	Miglioramento delle immagini	50
3.2.2	Estrazione delle caratteristiche	51
3.2.3	Classificazione e fusione	52
3.3	Risultati degli esperimenti	52
3.3.1	Database	52
3.3.2	Parametri	53
3.3.3	Risultati	56
4	Riconoscimento 3D. A computationally efficient approach to 3D ear recognition employing local and holistic features	61
4.1	Prerequisiti	62
4.1.1	Matrice di covarianza	62
4.1.2	Autofacce	63
4.2	Rappresentazione delle caratteristiche locali	64
4.2.1	Preprocessamento	64
4.3	Descrittore della caratteristica Histogram of Indexed Shapes (HIS)	64
4.3.1	Indice di forma e curvatura	65
4.3.2	Descrittore HIS	65
4.4	Individuazione dei keypoint 3D	66
4.5	Rappresentazione delle caratteristiche locali	69
4.5.1	Istogramma del taglio di superficie associata al descrittore di forma indicizzata (SPHIS)	70

4.6	Matching delle superfici locali	71
4.7	Estrazione della caratteristica olistica	75
4.7.1	Preprocessamento	75
4.7.2	Voxelizzazione della superficie	75
4.7.3	Motore di matching per la superficie olistica	77
4.7.4	Fusione	77
4.8	Risultati	78
4.8.1	Dati	78
4.8.2	Prestazioni di riconoscimento	79

Sommario

La possibilità di identificare le persone tramite l'orecchio esterno è stato introdotta per la prima volta dal criminologo francese Bertillon e rivisitata, successivamente, dal poliziotto Iannarelli che propose un primo sistema di riconoscimento basato su sette caratteristiche dell'orecchio. La struttura dettagliata dell'orecchio non è soltanto unica, ma, anche, non muta nel tempo. L'acquisizione dell'immagine dell'orecchio non è troppo intrusiva e non richiede la collaborazione da parte delle persone. Ecco perché negli ultimi anni lo studio di tale parametro biometrico ha trovato una notevole crescita. In questa tesi si vuole esporre il funzionamento di alcuni dei più recenti algoritmi per la registrazione delle immagini dell'orecchio e per il riconoscimento delle persone a partire da tali immagini precedentemente registrate.

Introduzione

L'individuazione di caratteristiche strutturali all'interno delle immagini è sempre stato un problema vitale ed impegnativo all'interno della "computer vision". L'utilizzo di tali caratteristiche può variare dal campo medico al campo dell'investigazione forense.

Nello specifico il parametro biometrico dell'orecchio esterno è stato adottato maggiormente per l'individuazione degli individui. Infatti mentre le password o ID card possono essere rubate o dimenticate, le caratteristiche biometriche sono universali, uniche, permanenti e misurabili [1].

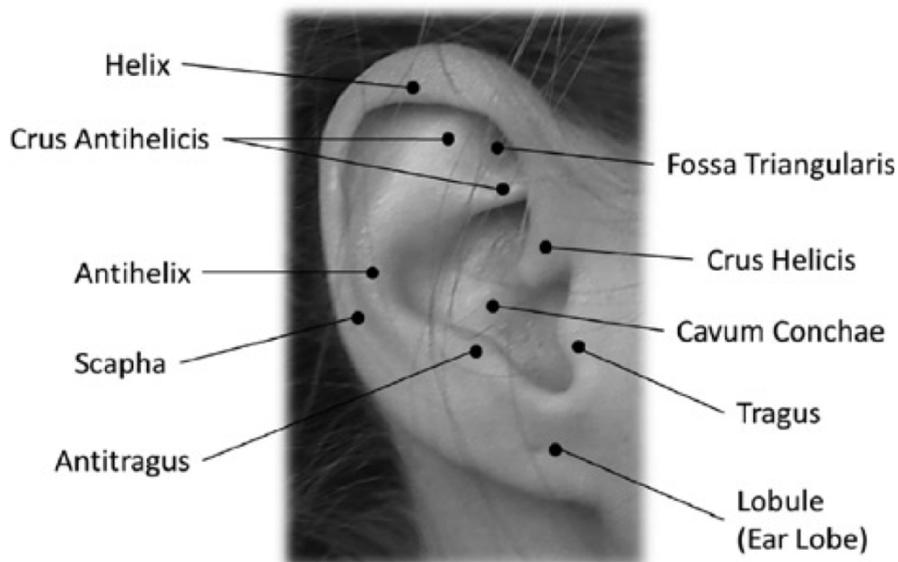


Figura 1: Orecchio esterno

L'orecchio esterno è formato dall'elica, dall'antielica, dal lobo, dal trago, dall'antitrago e dalla conca. La struttura dell'orecchio evolve dall'embrione fino al sesto mese d'età di vita di un individuo. La sua struttura non è completamente casuale, ma può essere suddivisa in parti. L'influenza dei

fattori casuali si può comunque vedere confrontando l'orecchio destro con l'orecchio sinistro della stessa persona; anche se simili le due orecchie non sono simmetriche [2].

La forma dell'orecchio esterno ha acquisito valore nell'identificazione delle persone nelle investigazioni criminali. Il criminologo francese Alphonse Bertillon fu il primo a rendersi conto dell'importanza dell'orecchio nelle investigazioni criminali più di un secolo fa [3]. Nel 1906 Richard Imhofer aveva bisogno di solo quattro caratteristiche per distinguere tra 500 differenti orecchie [4]. Dal 1949 il poliziotto americano Alfred Iannarelli concluse il primo studio su larga scala sul potenziale discriminativo dello orecchio. Raccolse 10.000 immagini d'orecchio e trovò 12 caratteristiche necessarie per riconoscere le persone senza ambiguità [5]. Iannarelli fece studi anche sulle orecchie di gemelli e trigemelli scoprendo che risultano uniche anche tra persone geneticamente identiche. Anche se gli studi di Iannarelli non avevano delle solide basi teoriche, si credeva ugualmente che la forma dell'orecchio esterno fosse unica. Tali studi costituirono un suggerimento e non un'evidenza dell'unicità della forma dell'orecchio esterno.

Gli studi sui cambiamenti della forma dell'orecchio dimostrarono che l'orecchio esterno cambia leggermente la sua dimensione con l'invecchiamento della persona [6, 7]. Infatti con l'età la struttura della cartilagine dell'orecchio cambia facendo ridurre l'elasticità della pelle dell'orecchio. Si sono effettuati degli studi sugli effetti del tempo sul riconoscimento dell'orecchio [8] dimostrando che le prestazioni di riconoscimento non sono affette dal trascorrere del tempo. Va ricordato che l'intervallo di tempo massimo considerato negli studi precedenti è di 10 mesi; ciò significa che sarebbe necessario uno studio più approfondito sugli effetti del tempo nel riconoscimento dell'orecchio esterno.

L'immagine dell'orecchio può essere facilmente catturata da distanza e senza la completa collaborazione della persona. Ciò rende il riconoscimento dello orecchio interessante per semplici sistemi di sorveglianza o per l'analisi forense di tali immagini. Ai giorni d'oggi l'osservazione delle sue caratteristiche è uno standard d'investigazione forense ed è usato come prova in centinaia di casi in diversi paesi europei (Forensic Ear identification Project [9]).

Attualmente sono stati sviluppati vari algoritmi che si diversificano in algoritmi di registrazione e algoritmi di riconoscimento. I primi servono, a partire dall'immagine iniziale, ad identificare ed enfatizzare il parametro biometrico che vogliamo considerare, mentre i secondi partono già da un insieme (un database) di immagini del parametro biometrico voluto ed individuano la persona associata. Gli algoritmi si dividono ulteriormente in tipologia 2D e 3D a seconda dell'immagine sorgente che utilizzano. In totale le tipologie

soggetto allo studio saranno quattro: riconoscimento 2D, registrazione 2D, riconoscimento 3D e registrazione 3D. Lo scopo della tesi è descrivere alcuni dei più recenti algoritmi per ognuna delle quattro tipologie.

Capitolo 1

Registrazione 2D. A novel ray analogy for enrolment of ear biometrics

Si descrive una tecnica di **registrazione delle immagini 2D** dell'orecchio esterno. Nel paper "A novel ray analogy for enrolment of ear biometrics" di Alistair, H., Cummings, A.H., Nixon, M.S., Carter, J.N. (2010) si introduce una nuova tecnica di registrazione a raggio dell'immagine basata sull'analogia con i raggi della luce.

La trasformata a raggio dell'immagine è una tecnica che estrae le caratteristiche tubolari e circolari dell'immagine. Essa usa una tecnica di tracciamento del raggio basato sui pixel che, attraverso le leggi dell'ottica, traccia i raggi all'interno dell'immagine. Tali raggi reagiscono a certe strutture fisiche enfatizzandole.

La trasformata è basata sui principi dell'ottica mentre i suoi dettagli sono definiti al fine di enfatizzare il successo nell'estrazione delle caratteristiche volute piuttosto che rappresentare con cura il fenomeno fisico. L'implementazione della trasformata considera solo le basi dell'analogia col fenomeno fisico e non si addentra nella simulazione dei dettagli della propagazione della luce. Sebbene la trasformata sia basata sull'analogia con la luce, essa può essere modificata al fine di enfatizzare ulteriormente le caratteristiche da individuare nell'immagine oppure per migliorare le prestazioni della trasformata stessa.

Al fine di comprendere al meglio il funzionamento dell'algoritmo proposto subito di seguito si andranno ad esplicitare dei concetti teorici dati per acquisiti dall'algoritmo proposto stesso.

1.1 Leggi dell'ottica

La propagazione della luce può essere simulata attraverso l'uso dei raggi. I raggi sono un modello per la propagazione delle onde, il più delle volte sotto forma di luce, che tiene conto della direzione dei raggi all'interno dell'ambiente che attraversa, senza dare importanza ad altri fenomeni legati alle onde come, ad esempio, la diffrazione.

1.1.1 Indice di rifrazione

L'indice di rifrazione n di un mezzo è il rapporto tra la velocità della luce nel vuoto c ($c = 300 \cdot 10^6$ m/s) e la velocità della luce nel mezzo v :

$$n = \frac{c}{v} \tag{1.1}$$

La tabella 1.1 riporta gli indici di rifrazione di alcune sostanze comuni.

Sostanza	Indice di rifrazione n
Solidi a 20° C	
Diamante	2,419
Ghiaccio (a 0° C)	1,309
Vetro (ottico)	1,523
Liquidi a 20° C	
Acqua	1,333
Alcol etilico	1,362
Gas a 0° C, 1 atm	
Aria	1,000293
Ossigeno	1,000139

Tabella 1.1: Indici di rifrazione misurati con una luce di lunghezza d'onda nel vuoto pari a 589 nm. [15]

L'indice di rifrazione dell'aria è così vicino a 1 che per la maggior parte degli scopi pratici si può porre $n_{aria} = 1$. In natura è un valore compreso tra $1 \leq n < 4$.

1.1.2 Legge della rifrazione

Quando la luce colpisce la superficie di separazione tra due mezzi diversi, in genere si divide in due parti, come mostra la figura Fig. 1.1 A: una parte viene riflessa con un angolo di riflessione uguale all'angolo di incidenza e una parte continua a propagarsi oltre la superficie di separazione. Se il raggio incidente non è perpendicolare alla superficie di separazione, il raggio che penetra nel secondo mezzo ha una direzione diversa da quella del raggio incidente. Nel nostro modello considereremo soltanto la possibilità che la luce o si rifranga o si rifletta totalmente internamente. Il raggio che penetra



Figura 1.1: Riflessione e rifrazione della luce fra due mezzi diversi m_1 e m_2 [15]

nel secondo materiale è rifratto e:

- quando la luce passa da un mezzo con indice di rifrazione minore (l'aria) a uno con indice di rifrazione maggiore (l'acqua), il raggio rifratto si avvicina alla normale (Fig. 1.1 A);
- quando la luce passa da un mezzo con indice di rifrazione maggiore (l'acqua) a uno con indice di rifrazione minore (l'aria), il raggio rifratto si allontana dalla normale (Fig. 1.1 B).

In entrambe le situazioni l'angolo di incidenza, l'angolo di riflessione e l'angolo di rifrazione sono misurati rispetto alla normale alla superficie di separazione nel punto di incidenza. Osserviamo che nella parte A della figura l'indice di rifrazione dell'aria è indicato con n_1 , mentre nella parte B è indicato con n_2 : il motivo di questa differenza è che si indicano con il pedice 1 tutte le variabili associate al raggio incidente (o al raggio riflesso) e con il pedice 2

tutte le variabili associate al raggio rifratto. L'angolo di rifrazione θ_2 dipende dall'angolo di incidenza θ_1 e dagli indici di rifrazione n_1 e n_2 dei due mezzi fisici. La relazione tra queste grandezze è chiamata legge della rifrazione di Snell che recita:

Quando la luce passa da un mezzo con indice di rifrazione n_1 a un mezzo con indice di rifrazione n_2 , il raggio incidente, il raggio rifratto e la normale alla superficie di separazione dei due mezzi nel punto di incidenza giacciono tutti nello stesso piano e l'angolo di rifrazione θ_2 è legato all'angolo di incidenza θ_1 dalla relazione:

$$n_1 \cdot \sin(\theta_1) = n_2 \cdot \sin(\theta_2) \quad (1.2)$$

1.1.3 Riflessione totale

Quando la luce passa da un mezzo fisico con indice di rifrazione maggiore a un mezzo fisico con indice di rifrazione minore (per esempio dall'acqua all'aria), il raggio rifratto si allontana dalla normale, come mostra la figura 1.2 A. Quando aumenta l'angolo di incidenza, aumenta anche l'angolo di rifrazione. Quando l'angolo di incidenza raggiunge un certo valore, chiamato angolo limite θ_L , l'angolo di rifrazione è di 90° . In questo caso il raggio rifratto è radente alla superficie di separazione, come mostra la parte B della figura 1.2. Quando l'angolo di incidenza è maggiore dell'angolo limite, il raggio rifratto manca: tutta la luce incidente viene riflessa all'interno del mezzo da cui proveniva (figura 1.2 C). Questo fenomeno è chiamato riflessione totale. Per ottenere il valore dell'angolo limite imponiamo che l'angolo di rifrazione sia $\theta_2 = 90^\circ$; dalla legge della rifrazione $n_1 \cdot \sin(\theta_1) = n_2 \cdot \sin(90^\circ)$ otteniamo $n_1 \cdot \sin(\theta_1) = n_2$, quindi l'angolo limite θ_L è tale che:

$$\begin{aligned} \sin(\theta_L) &= \frac{n_2}{n_1} \quad \text{con } (n_2 < n_1) \\ \theta_L &= \arcsin\left(\frac{n_2}{n_1}\right) \quad \text{con } (n_2 < n_1) \end{aligned} \quad (1.3)$$

1.2 Filtri

1.2.1 Filtri spaziali

Quando si applica un filtro spaziale ad un'immagine si esegue un'operazione di convoluzione discreta bidimensionale [17]. La formula che la rappresenta

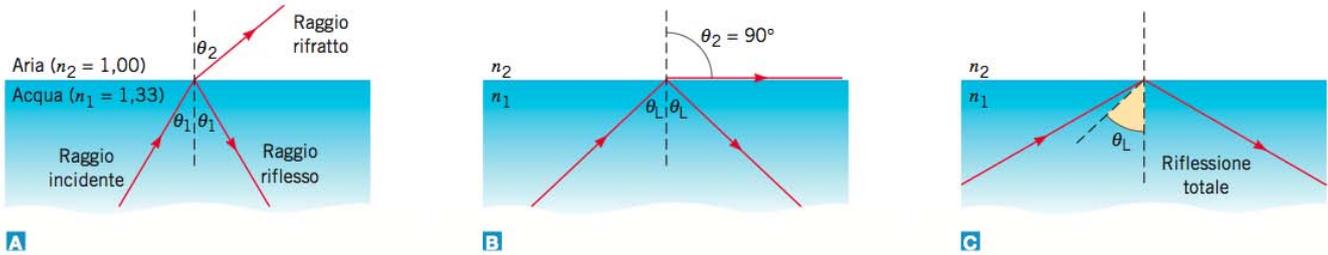


Figura 1.2: Riflessione totale della luce [15]

è:

$$g(i, j) = \sum_{l=0}^{M-1} \sum_{k=0}^{N-1} f(l, k) h(i-l, j-k) \quad (1.4)$$

con l'immagine di input $f(l, k)$, l'immagine di output $g(i, j)$ di dimensioni $M \times N$ mentre la risposta impulsiva $h(l, k)$ è sempre definita dalla matrice quadrata $L \times L$.

In questo caso la sommatoria dell'operatore di convoluzione ha componenti nulle nelle zone di non sovrapposizione tra le funzioni f ed h . In altre parole il dominio di interesse è definito dalle dimensioni della risposta impulsiva h . Per la proprietà commutativa dell'operatore di convoluzione, l'equazione precedente può essere espressa come segue:

$$g(i, j) = \sum_{l=0}^{M-1} \sum_{k=0}^{N-1} f(i-l, j-k) h(l, k) \quad (1.5)$$

Le ultime due equazioni dell'operatore di convoluzione suggeriscono che, prima del prodotto pixel per pixel, una delle due immagini, di input oppure della risposta impulsiva, deve essere ruotata di 180° rispetto all'origine e traslata.

Per convenienza l'immagine di input e di output sono scelte con identiche dimensioni, mentre la risposta impulsiva bidimensionale normalmente ha dimensioni limitate, a partire da matrici 3×3 proseguendo, eventualmente, con valori dispari.

Sempre per convenienza, il risultato $g(i, j)$ della convoluzione nell'immagine di output è associato al pixel centrale (i, j) della matrice quadrata in cui è definita la risposta impulsiva $h(l, k)$. Dall'equazione 1.4 si ricava che il valore $g(i, j)$ della convoluzione in corrispondenza del pixel (i, j) nell'immagine di output è ottenuto dalla somma del prodotto pixel a pixel tra la funzione di input $f(l, k)$ e la risposta impulsiva $h(i-l, j-k)$. $h(i-l, j-k)$ è derivata da $h(l, k)$ eseguendo prima una rotazione di 180° rispetto all'origine,

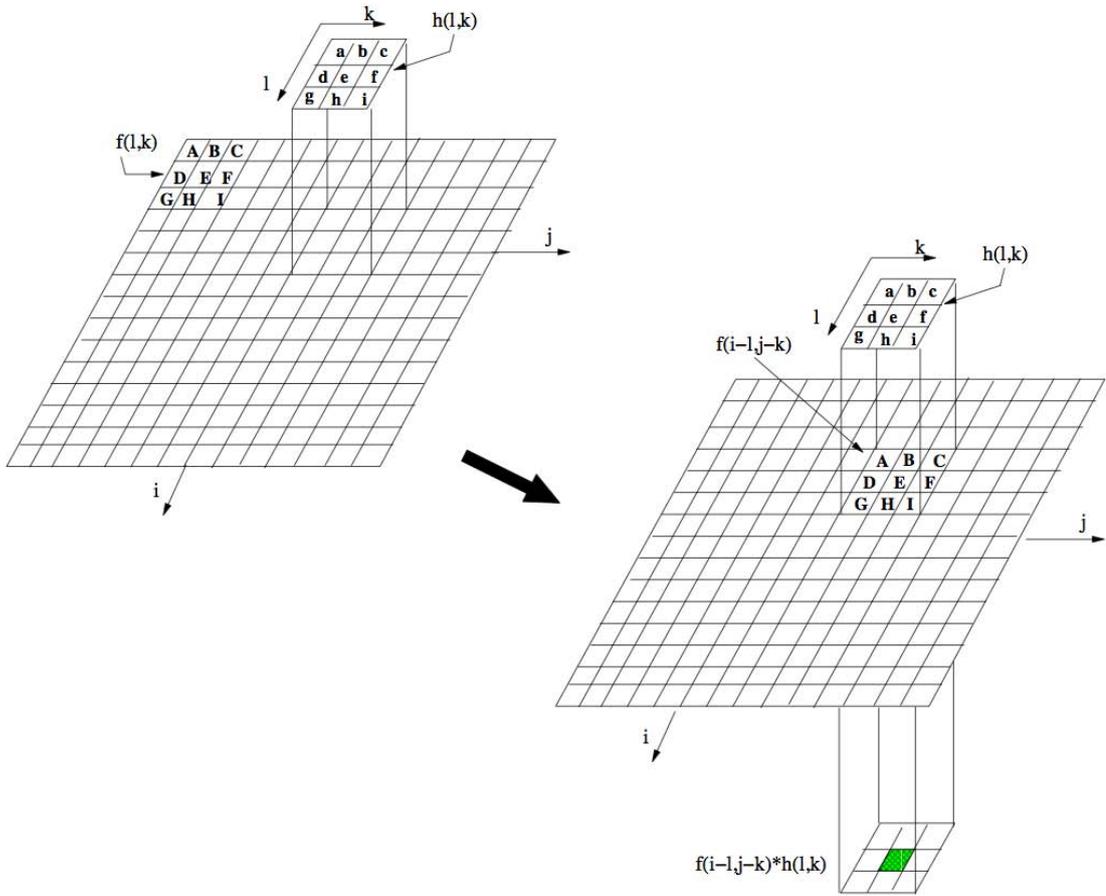


Figura 1.3: Convoluzione bidimensionale

ottenendo quindi $h(0-l, 0-k)$, e successivamente traslata dall'origine alla posizione (i, j) ottenendo $h(i-l, j-k)$ (in Fig. 1.3 si mostra il caso duale riferito all'equazione 1.5 in cui sono invertiti i ruoli fra h ed f).

Per ragioni implementative è conveniente utilizzare operatori di convoluzione con equazioni della forma:

$$g(i, j) = \sum_{l=-r}^r \sum_{k=-r}^r f(i-l, j-k)h(l, k) \quad (1.6)$$

con

- l'immagine della risposta impulsiva $h(l, k)$ con gli indici l e k riferiti rispetto al pixel centrale $h(0, 0)$.
- le dimensioni $L \times L$ dell'immagine della risposta impulsiva che hanno valore dispari, con $r = (L - 1)/2$.

- il valore della convoluzione $g(i, j)$ nel punto (i, j) che è calcolato solo per i pixel dell'immagine di input che si sovrappongono completamente con l'immagine della risposta impulsiva h .
- il pixel centrale $h(0, 0)$ dell'immagine della risposta impulsiva che è posizionato con il pixel $f(i, j)$ dell'immagine di input.

Consideriamo un esempio di convoluzione spaziale dove la risposta impulsiva è rappresentata da una matrice 3×3 con i pixel contenenti valori discreti di una funzione gaussiana. In questo caso il pixel centrale $h(0, 0)$ coincide con il picco della gaussiana; essendo tale funzione simmetrica rispetto all'origine, la rotazione o della funzione di input f o della stessa h non avrebbe alcun effetto. Il valore della convoluzione $g(i, j)$, applicata al pixel (i, j) è ottenuto posizionando il pixel centrale della maschera di convoluzione (rappresentata da f oppure h a seconda dell'equazione impiegata) in (i, j) , eseguendo l'espressione:

$$\begin{aligned}
 g(i, j) = & A \cdot f(i - 1, j - 1) + B \cdot f(i - 1, j) + C \cdot f(i - 1, j + 1) + \\
 & D \cdot f(i, j - 1) + E \cdot f(i, j) + F \cdot f(i, j + 1) + \\
 & G \cdot f(i + 1, j - 1) + H \cdot f(i + 1, j) + I \cdot f(i + 1, j + 1)
 \end{aligned} \tag{1.7}$$

dove $f(i, j)$ è l'immagine di input, le costanti da A ad I indicano i coefficienti della convoluzione ossia i valori discreti risultati dal campionamento della risposta impulsiva $h(i, j)$. Questa operazione di convoluzione è ripetuta per tutti i pixel dell'immagine di input $f(i, j)$ di dimensione $M \times N$.

In Fig. 1.5 è rappresentata la funzione di trasferimento h . La matrice contiene i valori campionati della risposta impulsiva discreta h dell'operatore che si vuole implementare. Occorre quindi calcolare i coefficienti $h(i, j)$ che rappresentano una buona approssimazione della funzione di trasferimento h e quindi dell'operatore che si vuole applicare. La convoluzione è un'operazione invariante perché i coefficienti della convoluzione (detti anche pesi del filtro) non cambiano da pixel a pixel durante il processo di convoluzione.

L'operatore di convoluzione per una maschera (o funzione di trasferimento) 3×3 e, per una immagine di 512×512 richiederebbe $226400 \cdot 9 = 2037000$ moltiplicazioni e 2037000 addizioni.

Vediamo alcuni aspetti implementativi della convoluzione. Quando l'operatore di convoluzione è applicato ai bordi dell'immagine, la maschera di convoluzione (o funzione di trasferimento) non si sovrappone completamente con i pixel dell'immagine. Per risolvere il problema o:

- si considerano a zero o uguali ad un valore costante tutti i pixel mancanti dell'immagine.

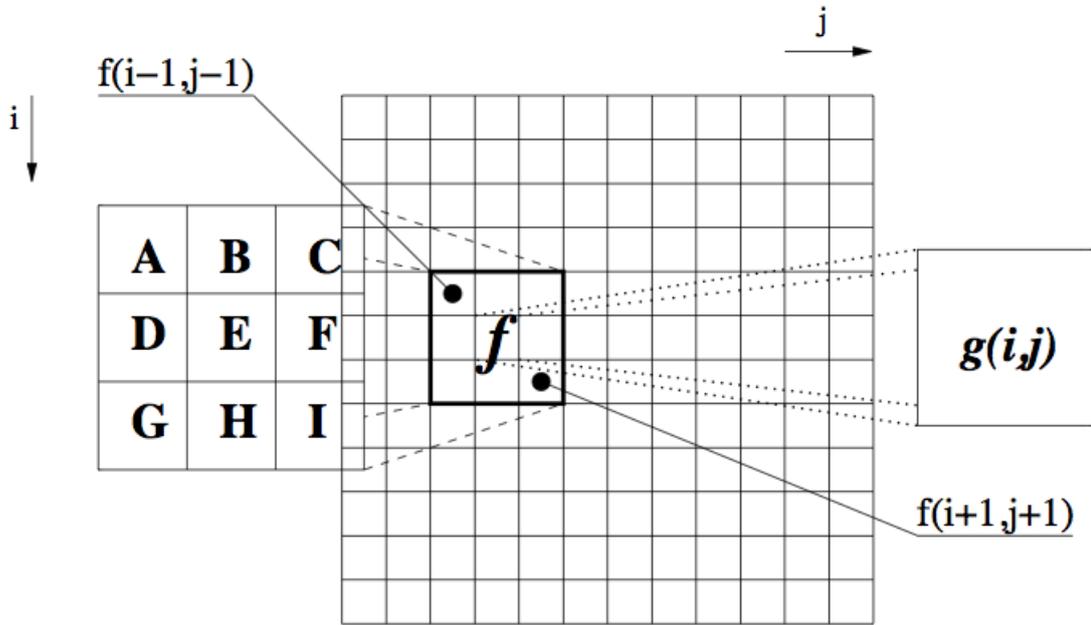


Figura 1.4: Schema dell'operazione di convoluzione

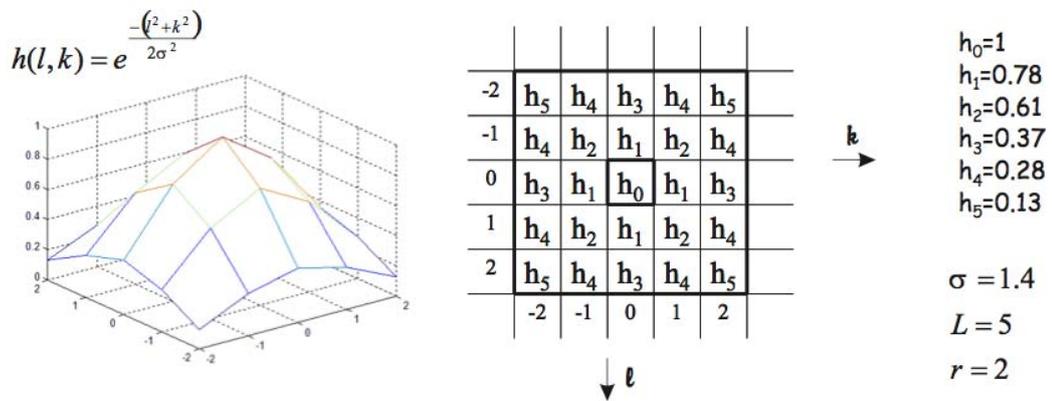


Figura 1.5: Funzione di trasferimento Gaussiana h

- si applica la convoluzione solo ai pixel dell'immagine che non creano il problema della mancanza di pixel.

La convoluzione produce come risultato una nuova immagine $g(i, j)$ che dovrà essere salvata in un'area di memoria separata dall'immagine di input f .

Riassumendo la nuova immagine g è ottenuta applicando, da sinistra verso destra e procedendo riga per riga, l'operazione di convoluzione come

schematizzato nella figura Fig. 1.6 e Fig. 1.7:

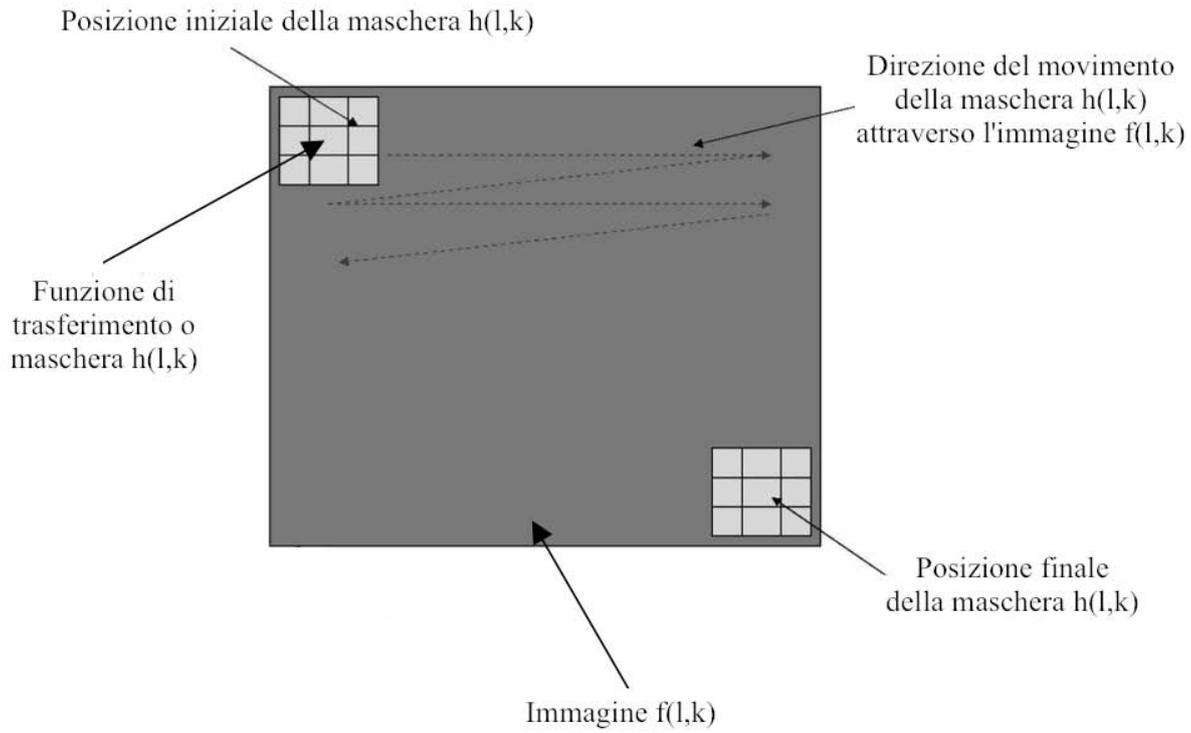


Figura 1.6: Applicazione di h all'immagine f

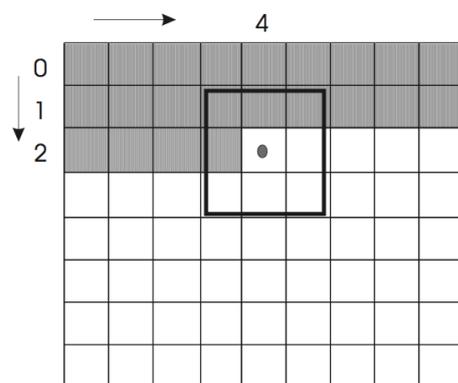


Figura 1.7: Nuova immagine g ottenuta dalla convoluzione

1.2.2 Filtro del gradiente

I filtri di edging devono esaltare le strutture ad alte frequenze ossia le discontinuità dei valori di intensità, ed eliminare le basse frequenze, ossia le piccole variazioni di valori di intensità.

L'operatore matematico che meglio evidenzia le discontinuità di una funzione è l'operatore di derivata prima. Per una immagine bidimensionale $f(x, y)$, le derivate parziali prime $\partial f/\partial x$ e $\partial f/\partial y$ definiscono le discontinuità dell'immagine nella direzione degli assi coordinati x e y .

L'orientazione dei bordi dell'immagine non necessariamente coincide con gli assi coordinati. Ne consegue la necessità di calcolare le derivate direzionali dell'immagine f per ogni punto (x, y) nella direzione r di massima discontinuità dei livelli di grigio (massimo locale della derivata di f) come rappresentato in Fig. 1.8. L'operatore che presenta queste caratteristiche è

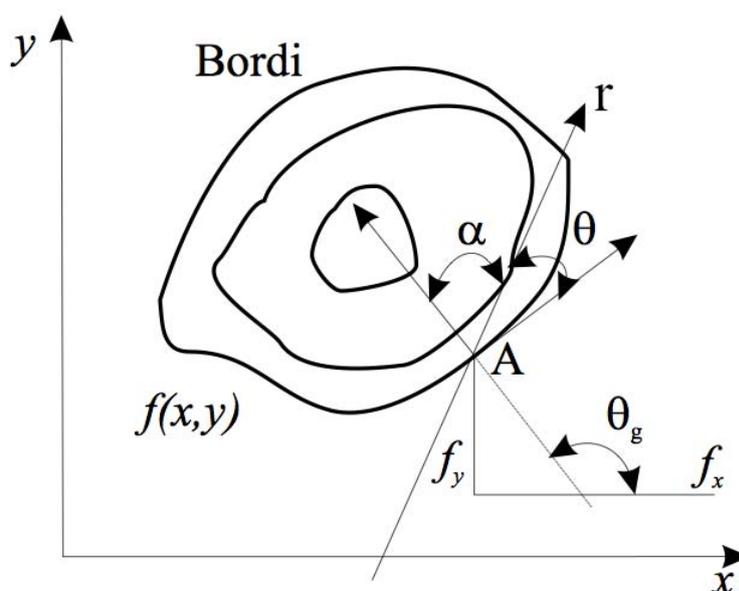


Figura 1.8: Profilo di una discontinuità in termini di gradiente e sua rappresentazione grafica

il gradiente. Il gradiente di $f(x, y)$ nel punto A , nella generica direzione r con angolo θ è dato da:

$$\frac{\partial f}{\partial r} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial r} = \frac{\partial f}{\partial x} \cos \theta + \frac{\partial f}{\partial y} \sin \theta \quad (1.8)$$

Siamo interessati al valore massimo del gradiente $\partial f/\partial r$, orientato nella direzione di massima variazione della funzione f . Questo è ottenuto derivando la

funzione gradiente rispetto a θ e ponendo uguale a zero (visto che un punto di massimo o minimo ha derivata nulla), come segue:

$$\frac{\partial}{\partial \theta} \left(\frac{\partial f}{\partial r} \right) = -\frac{\partial f}{\partial x} \sin \theta + \frac{\partial f}{\partial y} \cos \theta = 0 \quad (1.9)$$

da cui

$$\frac{\partial f}{\partial x} \sin \theta = \frac{\partial f}{\partial y} \cos \theta$$

$$\frac{\sin \theta}{\cos \theta} = \frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}} = \frac{f_y}{f_x}$$

$$\tan \theta = \frac{f_y}{f_x}$$

$$\theta_g(x, y) = \arctan \left(\frac{f_y}{f_x} \right) \quad (1.10)$$

dove $\theta_g(x, y)$ è la direzione di massima variazione di f (ossia la direzione perpendicolare al bordo) rispetto all'asse x , con f_x ed f_y le derivate parziali di f , rispettivamente, rispetto all'asse x ed all'asse y .

Viene d'ora in poi chiamato "vettore gradiente" il valore massimo della derivata $\frac{\partial}{\partial \theta} \left(\frac{\partial f}{\partial r} \right)$ nella direzione di massima variazione; in notazione differenziale è:

$$Grad(f) = \nabla f = \frac{\partial f}{\partial x} i + \frac{\partial f}{\partial y} j \quad (1.11)$$

dove i e j sono i vettori unitari degli assi coordinati x ed y rispettivamente. Una misura quantitativa della massima variazione di f nella direzione definita dal gradiente ∇f è data dal modulo del vettore gradiente definito come

$$|\nabla f| = \sqrt{f_x^2 + f_y^2} \quad (1.12)$$

Il modulo e la direzione del gradiente sono indipendenti dalle coordinate (x, y) del sistema di riferimento. Questo può essere dimostrato considerando che l'immagine f è una funzione scalare il cui valore dipende solo dal pixel A e non dalle sue coordinate (x, y) . Inoltre, se si considera in A , un vettore unitario \vec{s} , coincidente con la direzione r , risulta anche indipendente dalla scelta degli assi coordinati. Pertanto, la derivata $D_s(f)$ di f rispetto alla direzione r nella forma di prodotto interno tra vettori risulta:

$$D_s(f) = \frac{\partial f}{\partial r} = \vec{s} \cdot \nabla f = |\vec{s}| |\nabla f| \cdot \cos(\alpha) = |\nabla f| \cdot \cos(\alpha) \quad (1.13)$$

dove α è l'angolo tra il vettore unitario \vec{s} ed il vettore gradiente ∇f . Il valore della derivata $D_s(f)$ è massimo quando $\cos \alpha = 1$ che si verifica con $\alpha = 0$ e conseguentemente $D_s(f) = |\nabla f|$. Questo dimostra che il modulo e la direzione del gradiente di f sono indipendenti dalla scelta delle coordinate di riferimento.

1.2.3 Approssimazione del filtro gradiente

Per un immagine digitale $N \times N$ le componenti del vettore gradiente possono essere approssimate coi rispettivi rapporti incrementali nelle rispettive direzioni x e y .

Considerando le righe, per definizione di rapporto incrementale, si ha:

1) per la differenza in avanti si ha:

$$f_x(x, y) = \frac{\partial f}{\partial x}(x, y) = \lim_{h \rightarrow 0} \frac{f(x+h, y) - f(x, y)}{h}$$

2) per la differenza all'indietro si ha:

$$f_x(x, y) = \frac{\partial f}{\partial x}(x, y) = \lim_{h \rightarrow 0} \frac{f(x, y) - f(x-h, y)}{h}$$

3) per la differenza simmetrica si ha:

$$f_x(x, y) = \frac{\partial f}{\partial x}(x, y) = \lim_{h \rightarrow 0} \frac{f(x+h, y) - f(x-h, y)}{2h}$$

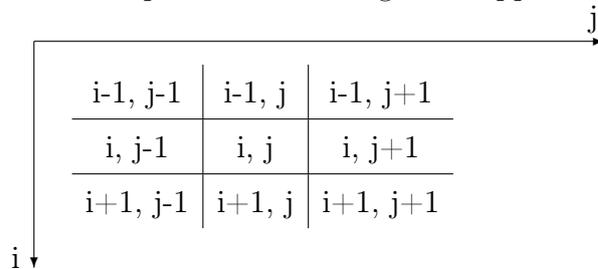
l'approssimazione voluta nei rispettivi casi è:

$$1) f_x[x, y] = \frac{\partial f}{\partial x}[x, y] = f[x+1, y] - f[x, y] = f[i, j+1] - f[i, j]$$

$$2) f_x[x, y] = \frac{\partial f}{\partial x}[x, y] = f[x, y] - f[x-1, y] = f[i, j] - f[i, j-1]$$

$$3) f_x[x, y] = \frac{\partial f}{\partial x}[x, y] = \frac{1}{2} (f[x+1, y] - f[x-1, y]) = \frac{1}{2} (f[i, j+1] - f[i, j-1])$$

L'ultima equivalenza nelle precedenti espressioni è data dalla convenzione di numerazione dei pixel di un immagine è rappresentata di seguito:



e i filtri differenziali o maschere di convoluzione, nei rispettivi casi, per calcolare il gradiente lungo le colonne sono:

$$1) h_C^+ = \begin{bmatrix} 1 \\ -1 \bullet \end{bmatrix}$$

$$2) h_C^- = \begin{bmatrix} 1 \bullet \\ -1 \end{bmatrix}$$

$$3) h_C^s = \frac{1}{2} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 1/2 \\ 0 \\ -1/2 \end{bmatrix}$$

il simbolo \bullet indica il pixel di elaborazione.

Le maschere di convoluzione orizzontale e verticale trovate sono utilizzate solo per l'estrazione dei bordi dominanti in direzione perpendicolare agli operatori stessi.

1.2.4 Filtro di Sobel

Gli operatori/filtri di estrazione dei bordi basati sul gradiente (es. filtro gradiente) generano bordi con spessore di almeno due pixel. Questi operatori sono sensibili al rumore dovuto alle piccole fluttuazioni dell'intensità dei pixel. Per attenuare questi inconvenienti si utilizza un operatore gradiente più complesso che, simultaneamente, esegue la differenziazione rispetto ad un asse coordinato e calcola una media locale in direzione ortogonale. Le componenti del gradiente, con l'operatore di Sobel, sono calcolate nella direzione degli assi coordinati x ed y , coinvolgendo, per ciascun pixel in elaborazione (i, j) , i pixel nelle sue vicinanze inclusi nella finestra 3×3 :

A	B	C
H	i, j	D
G	F	E

La differenziazione nella direzione x ed y è approssimata come segue:

$$f_x(i, j) = (C + KD + E) - (A + KH + G)$$

$$f_y(i, j) = (A + KB + C) - (G + KF + E)$$

dove la costante $K = 2$ ed i pixel vicini a quello in elaborazione (i, j) hanno una influenza corrispondente ai pesi indicati con A,B,...,E. Scegliendo una finestra più grande 3×3 per la stima del gradiente, l'effetto di smoothing aumenta rispetto ad un operatore con finestra 2×2 ; diminuisce anche la sensibilità dell'operatore alle fluttuazioni di intensità dell'immagine. Le maschere di convoluzione per l'operatore di Sobel sono:

$$h_R = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad h_C = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Si può notare come, nelle maschere, la somma dei pesi è uguale a zero. Da ciò conseguirà la non determinazione dei bordi in presenza di pixel con valori di grigio costante. I pesi nelle maschere sono appropriati per modellare un bordo ideale presente nella direzione degli assi coordinati. L'operatore attribuisce un peso maggiore ai pixel che sono più vicini al pixel (i, j) in elaborazione. Le convoluzioni discrete per le componenti orizzontale e verticale dell'operatore di Sobel sono:

$$g_R = f(i, j) * h_R(i, j) = \sum_{l=-1}^{+1} \sum_{k=-1}^{+1} f(i+l, j+k) h_R(l, k)$$

$$g_C = f(i, j) * h_C(i, j) = \sum_{l=-1}^{+1} \sum_{k=-1}^{+1} f(i+l, j+k) h_C(l, k)$$

L'operatore di Sobel definisce il modulo del gradiente con la formula nota:

$$|\nabla f| = \sqrt{g_R^2 + g_C^2}$$

1.3 Algoritmo proposto

Il funzionamento della trasformata a raggi dell'immagine (IRT) consiste nel tracciare un certo numero di raggi che attraversano un'immagine. Il percorso dei raggi ottenuto è usato per generare una nuova immagine che ne enfatizza le caratteristiche circolari o tubolari. L'immagine è associata ad un array di due dimensioni che rappresenta delle "celle di vetro". Ogni cella dell'array corrisponde ad un pixel dell'immagine; ogni cella contiene l'indice di rifrazione calcolato in relazione all'intensità i del pixel a cui è associata. Un metodo per calcolare l'indice di rifrazione di ogni pixel può essere o:

- attraverso una relazione lineare, con massimo indice di rifrazione n_{max} , data dall'equazione:

$$n_i = 1 + \left(\frac{i}{255} \right) \cdot (n_{max} - 1) \quad (1.14)$$

con i intensità del pixel nell'immagine e n_{max} massimo indice di rifrazione possibile.

- attraverso una relazione esponenziale, avente il parametro k che controlla la crescita dell'intensità i , data dall'equazione:

$$n_i = e^{\frac{i}{k}} \quad (1.15)$$

Gli indici di rifrazione vanno uniformemente da 1 a n_{max} .

All'interno dell'array è creato un raggio con una posizione casuale (x, y) di subpixel, direzione iniziale ϕ a partire da una funzione di distribuzione uniforme U . Per un'immagine di dimensione $w \times h$:

$$x \sim U[0, w), \quad y \sim U[0, h), \quad \phi \sim U[0, 2\pi), \quad (1.16)$$

Un vettore unitario \vec{v} che rappresenta la direzione del raggio è calcolato a partire da ϕ come:

$$\vec{v} = \begin{bmatrix} \cos(\phi) \\ \sin(\phi) \end{bmatrix} \quad (1.17)$$

Il raggio è inizializzato, (alla iterazione 0 come da apice), alla posizione x, y attraverso il vettore \vec{p} :

$$\vec{p}^{(0)} = \begin{bmatrix} x \\ y \end{bmatrix} \quad (1.18)$$

e all'iterazione t della trasformata si avrà:

$$\vec{\mathbf{p}}^{(t)} = \vec{\mathbf{p}}^{(t-1)} + \vec{\mathbf{v}} \quad (1.19)$$

La funzione raggio r traccia il cammino del raggio e aggiorna la matrice accumulatrice \mathbf{A} :

$$\mathbf{A}' = r(\mathbf{A}, \mathbf{I}, x, y, \vec{\mathbf{v}}, d, n_{max}) \quad (1.20)$$

Quando un raggio entra in un pixel per la prima volta la cella corrispondente della matrice accumulatrice \mathbf{A} è incrementata di 1:

$$\mathbf{A}'(\vec{\mathbf{p}}^{(t)}) = \mathbf{A}(\vec{\mathbf{p}}^{(t)}) + 1 \quad (1.21)$$

$\mathbf{A}(\vec{\mathbf{p}})$ è incrementato una volta sola per raggio in modo da evitare eventuali loop nel percorso del raggio che possono incrementare ripetitivamente un singolo pixel enfatizzando del rumore o altre caratteristiche indesiderate.

\mathbf{I} è la matrice le cui celle sono associate ai pixel dell'immagine. Per ogni cella si è calcolato l'indice di rifrazione del pixel e la normale $\vec{\mathbf{n}}$ alla direzione del bordo nel pixel calcolato con l'operatore Sobel.

d è il numero massimo di rifrazioni o riflessioni che il raggio può sopportare prima che il suo tracciamento termini: ciò è anche noto come profondità.

Quando il raggio attraversa la superficie di contatto di due pixel con diverso indice di rifrazione, deve essere calcolata nuovamente la direzione del raggio attraverso le leggi indicate nel paragrafo 1.1. Utilizzando il vettore direzione $\vec{\mathbf{v}}$, la normale $\vec{\mathbf{n}}$ (vettore unitario) e gli indici di rifrazione dei due mezzi n_1 e n_2 , si può calcolare la nuova direzione del raggio. Le equazioni per il calcolo della direzione per la rifrazione \mathbf{R}_r o per la riflessione \mathbf{R}_l a partire da $\vec{\mathbf{v}}$ ed $\vec{\mathbf{n}}$ sono calcolate come indicato in [16].

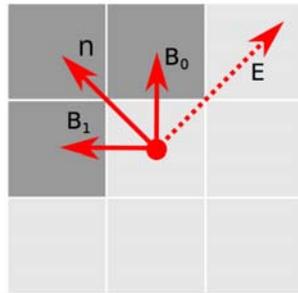


Figura 1.9: il vettore normale usato per il calcolo delle rifrazione o della riflessione

Se $\vec{\mathbf{n}}$ è la normale e n_1 e n_2 sono gli indici di rifrazione del primo e del secondo mezzo rispettivamente, l'angolo di incidenza θ_1 si trova da:

$$\cos(\theta_1) = \vec{\mathbf{n}} \cdot \vec{\mathbf{v}}$$

Se $n_1 > n_2$ bisogna testare se c'è rifrazione o riflessione interna. Se $n_2 > n_1$ o $\theta_1 < \theta_L$ assegnato a $z = \frac{n_1}{n_2}$, la direzione di \mathbf{R}_r è data da [16]:

$$\mathbf{R}_r = z\vec{\mathbf{v}} + (z(\vec{\mathbf{n}} \cdot \vec{\mathbf{v}}) - \cos(\theta_r))\vec{\mathbf{n}}$$

con

$$\cos(\theta_r) = \sqrt{1 - z^2(1 - \vec{\mathbf{n}} \cdot \vec{\mathbf{v}})}$$

Se $n_1 > n_2$ e $\theta_1 > \theta_L$ allora i raggi riflettono totalmente internamente. La direzione di \mathbf{R}_l è data da:

$$\mathbf{R}_l = \vec{\mathbf{v}} - 2(\vec{\mathbf{n}} \cdot \vec{\mathbf{v}})\vec{\mathbf{n}}$$

quindi \mathbf{R}_l o \mathbf{R}_r è assegnato a $\vec{\mathbf{v}}'$:

Il raggio è seguito in questa maniera finché rimane al di sotto delle d rifrazioni o riflessioni, finché ha percorso una certa distanza l o finché esce dall'immagine. Ciò è ripetuto o per N raggi o finché è raggiunta la condizione di stop e l'immagine trasformata è ottenuta dalla matrice accumulatore \mathbf{A} .

Mantenendo l'analogia coi raggi di luce, si fanno due modifiche al fine di migliorare i risultati della trasformata. La normale $\vec{\mathbf{n}}$ usata per il calcolo della nuova direzione del raggio non è definita verticalmente od orizzontalmente rispetto al bordo del pixel, ma è la normale alla direzione del bordo dell'immagine, ricavato dal filtro di Sobel, in quel pixel. Inoltre calcoleremo solo la rifrazione quando ci si muoverà da un indice di rifrazione più alto ad uno più basso; ciò migliora la qualità della trasformata ed enfatizza le caratteristiche strutturali.

In Fig. 1.9 la normale $\vec{\mathbf{n}}$ è ortogonale alla direzione del bordo della figura $\vec{\mathbf{E}}$ mentre $\vec{\mathbf{B}}_1$ e $\vec{\mathbf{B}}_2$ sono le ortogonali ai bordi del pixel.

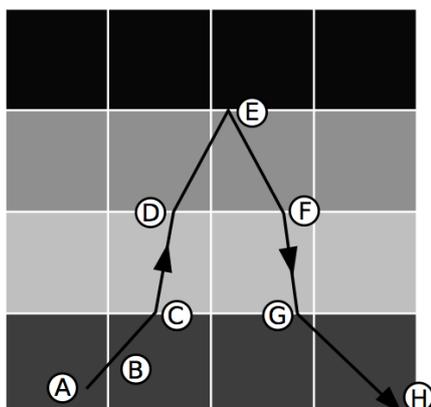


Figura 1.10: esempio del corso di una raggio in un immagine 4×4

La Fig. 1.10 mostra il percorso di un raggio che attraversa una semplice immagine. Più un pixel è chiaro maggiore è il suo indice di rifrazione. La direzione del bordo di ogni pixel \vec{E} è orizzontale quindi la norma \vec{n} è verticale. Il raggio entra nell'immagine alla posizione A con una direzione casuale e avanza verso B. In B sono comparati gli indici di riflessione del pixel corrente n_1 e del pixel successivo n_2 e visto che $n_1 = n_2$ il raggio continua senza cambio di direzione. In C $n_1 < n_2$ è così il raggio curva verso la normale, mentre in D $n_1 > n_2$ così il raggio si allontana dalla normale. In E vi è una riflessione totale interna visto che $n_1 > n_2$ e $\theta_1 = \theta_L$. In F e G avviene ciò che accade in C e D ma nel verso opposto mentre in H il raggio esce dall'immagine.

1.4 Parametri

La scelta dei parametri è stata effettuata per fare un compromesso fra eliminazione del rumore e prestazioni della trasformata. Vista la natura non deterministica della trasformata, dal momento che la direzione iniziale dei raggi è casuale, si devono comunque considerare un numero significativo di raggi che coprano tutte le caratteristiche dell'orecchio. Allo stesso tempo ogni passo dell'inseguimento di un raggio ha un costo computazionale e deve essere minimizzato.

Un parametro determinante è n_{max} o k a seconda della formula che si usa. Per evidenziare maggiormente le strutture dell'orecchio abbiamo usato valori che vanno $4 < n_{max} < 40$ oppure $5 < k < 100$.

I parametri l e d sono legati alla scala dell'immagine e alla scala delle caratteristiche da estrarre. d deve essere un numero tale che consenta al

raggio di seguire le strutture che deve enfatizzare, ma non troppo elevato visto che appesantirebbe la trasformata. In generale $d = 256$

Il parametro l viene usato per quelle immagini che hanno ampie parti con la stessa intensità e quindi stesso indice di rifrazione. In questi casi non avvengono molte variazioni nella direzione dei raggi e il percorso dei raggi termina quando l è stata percorsa. Si sceglie l lungo il doppio della lunghezza della diagonale dell'immagine in modo tale che sia improbabile il superamento di l prima di d .

N è il numero dei raggi che deve essere appropriato per coprire la superficie dell'immagine ma non deve essere eccessivo (appesantirebbe la trasformata). Dall'esperienza euristica si è visto che $N = 2000$ è sufficiente per un'immagine fino a 512×512 pixel. Risulta, comunque, preferibile definire una condizione di stop, come si vedrà di seguito, per stabilire quando fermare la trasformata.

La condizione di stop automatico della trasformata è ottenuta calcolando la differenza dell'accumulatore normalizzato durante le iterazioni; l'arresto avviene quando non c'è un significativo cambiamento nell'accumulatore ottenuto. La condizione di arresto si ha grazie all'errore quadratico medio della differenza delle intensità dell'accumulatore normalizzato a diverse iterazioni:

$$D^{<t>}(\mathbf{I}'^{<t>}, T) = \sqrt{\frac{1}{|\mathbf{I}'|} \sum_{z \in \mathbf{I}'} (\mathbf{I}'^{<t>}(z) - \mathbf{I}'^{<t-T>}(z))^2}$$

dove T è il numero di iterazioni che intercorrono prima di valutare la differenza e $\mathbf{I}'^{<t>}(z)$ è l'accumulatore normalizzato all'iterazione t . Visto che la normalizzazione è molto più lenta rispetto ad un'iterazione della trasformata è meglio scegliere un valore ampio per la T .

Il valore calcolato D è utilizzato come condizione di stop quando esso supera la soglia D_S . Solitamente $D_S = 1$ con $T = 1000$ e corrisponde alla variazione media di 1 nell'intensità del pixel ogni 1000 iterazioni. Lo pseudo codice per evidenziare la condizione di stop è:

```
while D>DS
  accumulator ← CastRadomRay(image, accumulator)
  if i % T = 0 then
    previousNormImage ← currentNormImage
    currentNormImage ← NormaliseArray(accumulator)
    D ← 0
    for z ← 0 to Size(image) do
      D ← D + (currentNormImage[z] - previousNormImage[z])2
    D ← SquareRoot(D/Size(Image))
  i ← i++
```

Listing 1.1: Pseudocodice della condizione di arresto

1.5 Risultati

L'individuazione è stata eseguita su 252 immagini (4 per individuo) da un database XM2VTS [19] applicando la trasformata Fig. 1.11 b), alcuni filtri successivi Fig. 1.11 c) e d) e una rotazione ed una scala finali Fig. 1.11 f).

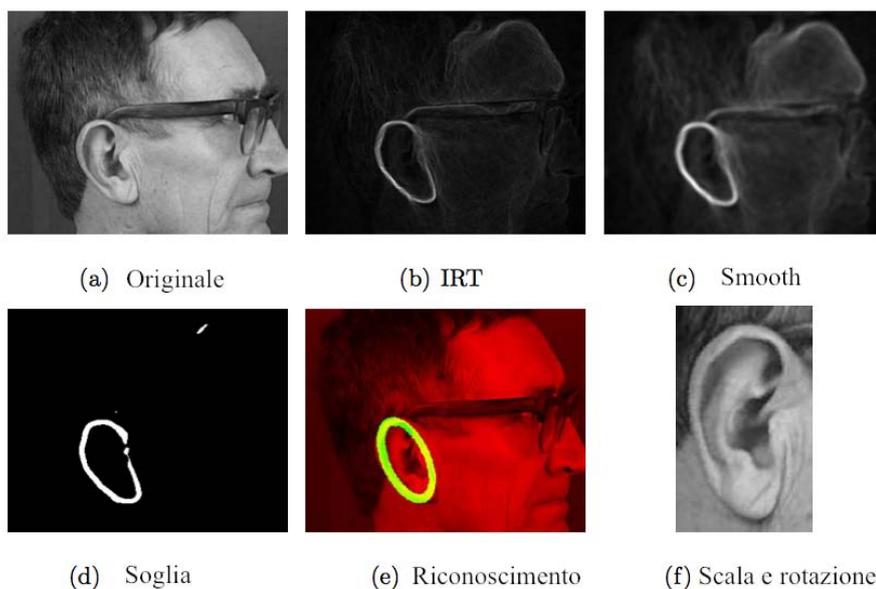


Figura 1.11: Passi per l'individuazione dell'orecchio

La computazione media di tutti i passi richiede 5,45s mentre l'IRT richiede 1,47s e richiede 19000 raggi per tracciare le caratteristiche dell'orecchio estero.

Su 252 immagini il 99,6% dell'immagini estratte conteneva l'orecchio. Il 98,4% delle immagini presentavano l'orecchio correttamente normalizzato ma con un errore nella scala o nella rotazione.

Capitolo 2

Registrazione 3D. An efficient technique for ear detection in 3D: Invariant to rotation and scale

Il paper “An efficient technique for ear detection in 3D: Invariant to rotation and scale” di Prakash, S. e Gupta, P. descrive una tecnica per la **registrazione delle immagini 3D** dell’orecchio esterno. Il metodo proposto si basa sul fatto che nel profilo di una faccia 3D, l’orecchio esterno è la sola zona che contiene le massime discontinuità di profondità. Inoltre i bordi appartenenti all’orecchio esterno sono curvi. La tecnica si divide in tre principali passi: pre-processamento, calcolo degli insiemi di orecchie candidate e localizzazione dell’orecchio.

Al fine di comprendere al meglio il funzionamento dell’algoritmo proposto subito di seguito si andranno ad esplicitare dei concetti teorici dati per acquisiti dall’algoritmo proposto stesso.

2.1 Filtri

2.1.1 Operatore derivata della Gaussiana (DroG)

L’operatore gradiente basato sulla derivata della gaussiana è ottenuto applicando prima l’operatore di smoothing gaussiano e successivamente derivando rispetto agli assi coordinati.

La gaussiana è una funzione a simmetria rotazionale la cui equazione nel caso 2D continuo è la seguente:

$$h(i, j) = e^{-\frac{i^2+j^2}{2\sigma^2}}$$

Il valore di σ determina l'apertura della gaussiana che aumenta col crescere di σ . Per quanto riguarda l'operatore DroG le risposte impulsive lungo le due direzioni, che lo caratterizzano completamente, si possono ottenere attraverso il calcolo della derivata della gaussiana rispetto a i e j che rappresentano le componenti dell'operatore DroG nel caso continuo. La risposta impulsiva di questo operatore, lungo l'asse orizzontale, risulta:

$$h_R(i, j) = \frac{\partial}{\partial i} e^{-\frac{i^2+j^2}{2\sigma^2}} = -\frac{i}{\sigma^2} e^{-\frac{i^2+j^2}{2\sigma^2}}$$

con σ la deviazione standard. Analogamente può essere fatto per calcolare la risposta impulsiva rispetto all'asse verticale:

$$h_C(i, j) = \frac{\partial}{\partial j} e^{-\frac{i^2+j^2}{2\sigma^2}} = -\frac{j}{\sigma^2} e^{-\frac{i^2+j^2}{2\sigma^2}}$$

Questo operatore rappresenta un esempio combinato tra filtro gaussiano di smoothing e l'operatore gradiente basato sulla derivata prima. Inoltre fornisce il miglior compromesso tra l'attenuazione del rumore (filtro di smoothing) e l'esaltazione dei bordi (operatore derivata).

2.1.2 Operatore di Canny

L'operatore di Canny è un operatore lineare basato sulla derivata prima della Gaussiana DroG che risulta essere un buon compromesso tra localizzazione dei bordi e non dipendenza dal rumore.

Questo operatore combina il filtraggio Gaussiano e la stima del gradiente, (basato sulle differenze finite), ed è spazialmente non isotropo. Le caratteristiche dell'operatore di Canny sono:

- *Buona determinazione dei bordi*: il rapporto segnale rumore (SNR) del gradiente è massimizzato per ottenere la minore probabilità di errore nella determinazione di un bordo reale e per ottenere la minore probabilità di individuazione di un falso bordo.
- *Buona localizzazione*: i punti identificati come bordi sono il più vicino possibile al centro del bordo reale.
- *Risposta singola*: l'operatore dovrebbe produrre una unica risposta per uno stesso bordo.

Esprimiamo matematicamente le caratteristiche sopra citate.

La prima e la seconda caratteristica possono essere soddisfatte considerando l'operatore DroG, la Derivata della Gaussiana, come operatore che

meglio approssima il rapporto segnale rumore e localizzazione. Il primo passo dell'operatore di Canny è quello di convolvere l'immagine $f(i, j)$ con un filtro di smoothing Gaussiano $h(i, j; \sigma)$ producendo una immagine $g(i, j)$ data da:

$$g(i, j) = f(i, j) * h(i, j; \sigma)$$

dove σ controlla il livello di smoothing. Successivamente sono stimate le derivate parziali g_x e g_y e il modulo ed orientazione del gradiente per ogni pixel (i, j) dell'immagine g . Le derivate parziali sono approssimate, usando per ogni pixel (i, j) finestre 2×2 , e considerando le differenze prime:

$$g_x(i, j) \cong [(g(i, j+1) - g(i, j)) + (g(i+1, j+1) - g(i+1, j))]/2$$

$$g_y(i, j) \cong [(g(i, j) - g(i+1, j)) + (g(i, j+1) - g(i+1, j+1))]/2$$

Il modulo ed orientazione del gradiente sono calcolati dalle note formule di conversione tra coordinate cartesiane e polari:

$$M(i, j) = \sqrt{g_x^2 + g_y^2}$$

$$\theta(i, j) = \arctan \left(\frac{g_y(i, j)}{g_x(i, j)} \right)$$

Dei vari $\theta(i, j)$ ci interessa solo la direzione e non il verso. Inoltre si posso approssimare i valori degli angoli ottenuti, per ogni pixel, al primo multiplo di 45° più vicino ($0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ$).

Il modulo $M(i, j)$ del gradiente presenta delle ampie rampe intorno ai massimi locali che sono in corrispondenza di bordi. Quindi applicare un semplice filtro di soglia non è sufficiente a stabilire la posizione precisa dei bordi. Per il calcolo della locazione del bordo si usa un processo è chiamato "soppressione dei non massimi locali" (non-maximal suppression, NMS). Tale processo assottiglia i picchi in $M(i, j)$ ponendo a zero tutti i punti di $M(i, j)$ lungo la direzione $\theta(i, j)$ del gradiente che non sono valori di picco. Precisamente la strategia adottata per identificare i massimi locali, consiste nel sopprimere i non-massimi basandosi sull'idea che ogni pixel centrato in un bordo avrà un'intensità superiore rispetto ai 2 pixel vicini nella direzione del gradiente. Si divide l'intorno di un pixel in 4 regioni (giallo, verde, rosso e blu) corrispondenti a 4 direzioni come mostrato in Fig. 2.1. Si verifica in che regione cade la direzione $\theta(i, j)$ associata al modulo $M(i, j)$ vettore gradiente del pixel in centro. Se l'intensità del pixel centrale $I(i, j)$ è inferiore all'intensità di almeno uno dei 2 pixel vicini, lungo la direzione del gradiente, il pixel viene soppresso. Mantenendo l'esempio della Fig. 2.1 si ha la seguente schematizzazione tabellare della matrice modulo del gradiente $M(i, j)$ associata all'intorno del pixel $I(i, j)$:

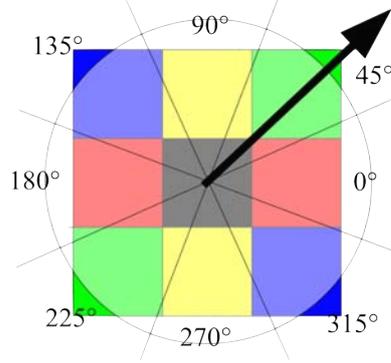


Figura 2.1: Divisione dell'intorno di un pixel in 4 regioni evidenziate dai colori giallo, verde, rosso e blu.

*	*	$M(i-1, j+1)$
*	$M(i, j)$	*
$M(i+1, j-1)$	*	*

quindi

$$I(i, j) = \begin{cases} M(i, j) & \text{se } M(i, j) \geq M(i+1, j-1) \text{ e } M(i, j) \geq M(i-1, j+1), \\ 0 & \text{altrimenti.} \end{cases}$$

Alla fine di questo processo, applicandolo per ogni pixel, otteniamo una nuova immagine $N(i, j)$. Tutte le zone in $M(i, j)$ con massimi locali sono state assottigliate con picchi di un solo pixel, producendo in output l'immagine:

$$N(i, j) = nms\{M(i, j); \theta(i, j)\}$$

L'immagine non-maximal suppression $N(i, j)$ contiene comunque diversi falsi bordi dovuti al rumore ed alla alta tessitura presente nell'immagine originale f anche se inizialmente smussata con filtro Gaussiano $h(i, j; \sigma)$. Un metodo per ridurre i falsi frammenti dei bordi presenti in $N(i, j)$, l'immagine modulo del gradiente (NMS), è quello di applicare una soglia adeguata T . Si è visto che usare una sola soglia ha il problema di lasciare falsi bordi se considerata troppo bassa e rimuovere bordi reali se presa troppo elevata. L'algoritmo di Canny utilizza 2 soglie $T_L < T_H$ in modo tale che $2 \leq \frac{T_H}{T_L} \leq 3$. Si calcolano, poi, 2 immagini: una di bordi "forti" e una di bordi "deboli". Tutti i valori di $N(i, j)$ al di sotto della soglia T_H sono cambiati a zero:

$$N_H(i, j) = N(i, j) \geq T_H$$

e tutti i valori di $N(i, j)$ non compresi tra la soglia T_L e T_H sono cambiati a zero:

$$N_L(i, j) = T_L \leq N(i, j) < T_H$$

A seconda del valore della soglia T_H i bordi presentano dei buchi. Per ottenere bordi più lunghi si considerano bordi validi anche tutti quei pixel marcati in N_L che sono connessi direttamente o tramite una catena a un pixel marcato in N_H come mostrato in Fig. 2.2.

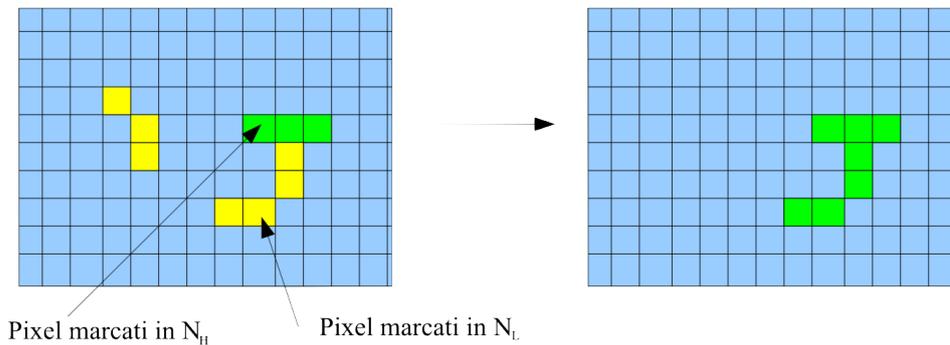


Figura 2.2: Pixel marcati nelle 2 immagini sogliate N_H e N_L e uniti se connessi

Riassumendo l'**algoritmo di Canny** si riassume in:

1. L'immagine originale $f(i, j)$ è convoluta con il filtro Gaussiano $h(i, j; \sigma)$; si ottiene così l'immagine smussata $g(i, j)$.
2. Per ogni pixel dell'immagine smussata $g(i, j)$ si calcolano:
 - $g_x(i, j)$: derivata prima parziale (componente orizzontale).
 - $g_y(i, j)$: derivata prima parziale (componente verticale).
 - $M(i, j)$: modulo del gradiente.
 - $\theta(i, j)$: direzione del gradiente.

Dal momento che il calcolo riguarda tutti i pixel di $g(i, j)$, il risultato del calcolo forma 4 immagini intermedie.

3. Si applica la procedura di non-maximal suppression (NMS) all'immagine modulo del gradiente $M(i, j)$ ottenendo come risultato l'immagine $N(i, j)$. Quest'ultima contiene i bordi localizzati lungo il gra-

diente ed i valori dei pixel rappresentano la robustezza dei bordi presenti nell'immagine originale $f(i, j)$ espressa in termini di modulo del gradiente.

4. Si eseguono delle euristiche per individuare le soglie adeguate per l'immagine $N(i, j)$ allo scopo di attenuare la presenza di falsi bordi e di collegare contorni interrotti.
5. (Fuori dall'algoritmo) Ripetere eventualmente i 4 passi precedenti con valori crescenti di σ_k ottenendo le immagini di bordo $N(i, j; \sigma_k)$ per avere diversi risultati dell'operatore di Canny.

2.2 Pre-processamento

Si suddivide in quattro passi. Al primo passo l'immagine 3D del profilo della faccia dell'individuo è convertita in un'immagine di profondità. Successivamente viene effettuato il calcolo dei bordi sull'immagine di profondità. Poi vengono approssimati i bordi ottenuti sostituendovi dei segmenti. Ed infine vengono eliminati i bordi irrilevanti.

2.2.1 Calcolo dell'immagine di profondità

L'immagine 3D del profilo di un individuo è ottenuta dallo scanner laser 3D digitizer Minolta Vivid 910 il quale produce dati scansionati 3D sotto forma di una griglia di punti $m \times n$ ognuno dei quali ha l'informazione sulla profondità 3D. Se lo scanner laser fallisce nel determinare la profondità di un punto assegna un valore ∞ alla profondità di tale punto. Se lo scanner determina la profondità assegna un valore $z(i, j)$ finito al punto (i, j) .

L'immagine di profondità $I_{2D} \in R^{m \times n}$ è ottenuta considerando il valore di profondità 3D di ogni pixel come valore di intensità proprio del pixel stesso:

$$I_{2D}(i, j) = \begin{cases} z(i, j) & \text{se } z(i, j) \text{ è finito,} \\ 0 & \text{altrimenti.} \end{cases}$$

L'intensità dei pixel nell'immagine di profondità è normalizzata per valori tra 0 e 1.

2.2.2 Calcolo dei bordi

Nell'immagine di profondità il calcolo dei bordi dell'orecchio è semplice, in quanto nella regione dell'orecchio vi è una forte discontinuità di profondità

in particolare nella zona dell'elica dell'orecchio. Inoltre l'orecchio esterno è la regione che contiene il maggior numero di bordi.

Nella tecnica proposta per calcolare i bordi all'immagine di profondità viene applicato l'algoritmo di Canny. Così è ottenuta una lista di tutti i bordi trovati connettendo i pixel appartenenti ai bordi assieme all'interno di un elenco di coppie di coordinate. Per ogni giunzione fra bordi che si riscontra l'elenco di coppie associato ad un bordo è terminato e ne viene creato uno separato per ogni ramo della giunzione riscontrato. I bordi ottenuti dal rumore sono eliminati usando un criterio di lunghezza del bordo in funzione dei pixel che contiene tale bordo. Per un'immagine larga n la soglia è $\tau = kn$ con k scelta sperimentalmente (es. $\tau = 0.03$).

2.2.3 Approssimazione dei bordi

Non necessariamente tutti i pixel appartenenti ai bordi finora calcolati sono necessari per rappresentare tali bordi. Per velocizzare il processo i pixel ridondanti sono eliminati rendendo più sottili i bordi trovati. Per ogni punto in una lista di un bordo si calcola la deviazione che c'è con la linea che congiunge i punti finali del bordo stesso. Per ogni lista si calcola il segmento che congiunge i due estremi del bordo; data una certa deviazione voluta si calcola la posizione di massima deviazione attorno a tale segmento. Se un pixel nella lista supera la tolleranza voluta il bordo viene accorciato nella posizione del pixel e il procedimento viene ripetuto mantenendo come nuovo limite del segmento la posizione del pixel. Questo processo spezza i bordi in segmenti con la tolleranza voluta.

2.2.4 Eliminazione dei bordi irrilevanti

Se dopo il passo precedente si sono ottenuti dei bordi con solamente due punti è poco probabile che appartengano alla struttura del viso umano. È necessario che i bordi ottenuti dopo il passo precedente abbiano almeno 3 punti che li rappresentano. Dopo l'eliminazione dei bordi non probabili si intende per S l'insieme dei bordi che possono appartenere all'orecchio esterno.

2.3 Calcolo dell'insieme candidato dell'orecchio

Il processo di calcolo dell'insieme candidato a rappresentare l'orecchio esterno si divide in: creazione del grafo di connettività dei bordi, calcolo dei componenti connessi nel grafo e calcolo dell'insieme candidato dell'orecchio.

2.3.1 Costruzione del grafo di connettività dei bordi

Sia S l'insieme che contiene n bordi che definiscono i contorni dell'immagine di profondità. Sia e_i i -esimo bordo in S ad essere definito dal punto $p_i \in P$. Ci sia un inviluppo convesso¹ $CH(e_i)$ definito per ogni e_i . Ci sia un grafo di connettività $G = (V, E)$ con

$$V = \{p_i \mid p_i \in P\}$$

$$E = \{(p_i, p_j) \mid CH(e_i) \text{ interseca } CH(e_j)\}$$

Si può notare che l'orecchio esterno ha superfici per lo più convesse. Quindi se ci si sposta dai bordi esterni dell'orecchio verso l'interno la maggior parte dei bordi esterni incontreremo anche i bordi interni. Per questo motivo l'inviluppo convesso dei bordi esterni conterrà al suo interno anche i bordi interni. É stato sperimentalmente provato che l'inviluppo convesso del bordo di un orecchio taglia almeno un altro inviluppo convesso di un altro bordo appartenente all'orecchio. Così ci si aspetta che tutti i vertici di G appartenenti all'orecchio siano connessi direttamente o attraverso altri vertici. Si può anche notare che la caratteristica che un bordo ne contenga altri appartiene solo alla zona dell'orecchio e non è presente in altre parti dell'immagine del profilo dell'individuo. Quindi i vertici non appartenenti all'orecchio rimangono isolati nel grafo.

2.3.2 Calcolo delle componenti connesse

Due vertici sono nella stessa componente connessa se esiste un percorso che li unisce. Componenti con due o più vertici probabilmente appartengono ad un probabile orecchio candidato. Componenti con vertici singoli possono essere eliminate in quanto non rappresenteranno mai la struttura dell'orecchio. La Fig. 2.3 mostra un immagine dei bordi ottenuti dal profilo di un individuo. La Fig. 2.4 mostra il grafo di connettività dei bordi ottenuto a partire dalla Fig. 2.3. Le componenti connesse in Fig. 2.4 con più di un vertice sono inserite in un rettangolo.

2.3.3 Calcolo degli orecchi candidati

Sia $K = \{K_1, K_2, \dots, K_m\}$ l'insieme dei componenti connessi nel grafo G dove ogni componente ha 2 o più vertici. Il grado medio del vertice di una

¹Dato un insieme di punti l'inviluppo convesso è il più piccolo insieme convesso che li contiene tutti.



Figura 2.3: Mappa dei bordi

componente connessa K_j è dato da:

$$d(K_j) = \frac{\sum_{i=1}^{n_j} d(p_i)}{n_j} \quad (2.1)$$

dove $d(p_i)$ è il grado del vertice p_i e n_j è il numero dei vertici in K_j . Dal momento che l'orecchio ha molti bordi dovuti alla notevole variazione di profondità dei pixel, è poco probabile che componenti connesse, aventi solo due vertici o grado di vertice medio 1, rappresentino un orecchio. Quindi solo i componenti con grado di vertice medio maggiore di 1 sono considerati come probabili bordi dell'orecchio. Un probabile orecchio candidato è definito come quella parte dell'immagine, del profilo dell'individuo, compresa all'interno di un ritaglio rettangolare effettuato nel grafo di connettività che contiene bordi connessi. Un insieme di orecchie candidate è calcolato usando tutti i componenti connessi che soddisfino il criterio del grado di vertice medio.

2.4 Localizzazione dell'orecchio

Viene effettuata trovando il vero orecchio fra i candidati attraverso l'uso di un modello d'orecchio.

2.4.1 Creazione del modello d'orecchio

Per calcolare il modello è usato un descrittore di forma 3D detto *distribuzione di forma* [28] che è invariante alla rotazione e alla scala. La distribuzione di forma fornisce caratteristiche distintive fra orecchie e non orecchie ed è

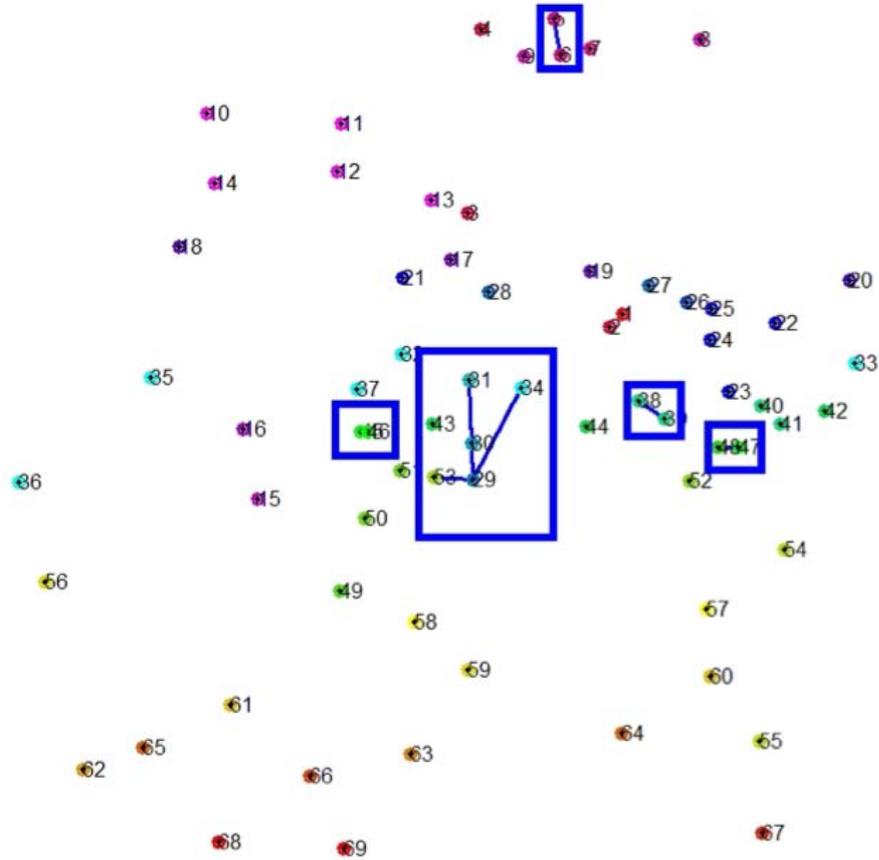


Figura 2.4: Grafo di connettività dei bordi

robusta al cambiamento dell'angolo di visione dell'immagine, alla rotazione e alla scala. La distribuzione di forma rappresenta la forma di un modello 3D come la distribuzione di probabilità campionata da una *funzione di forma* che misura le proprietà geometriche del modello 3D. Useremo la funzione norma Euclidea come funzione di forma per calcolare la distribuzione di forma. La distribuzione di forma rappresenta la distanza Euclidea fra coppie di punti scelti a caso sulla superficie del modello 3D. La distribuzione descrive la forma complessiva dell'oggetto in modo efficiente ed i campionamenti si possono ottenere facilmente e velocemente dalla distribuzione. In Fig. 2.6 è rappresentata la distribuzione di forma di 200 campioni d'orecchio e altrettanti di non-orecchio. Chiaramente la distribuzione di forma delle orecchie è la parte rossa del grafico.

Quindi per calcolare il modello d'orecchio si parte da un insieme di trai-

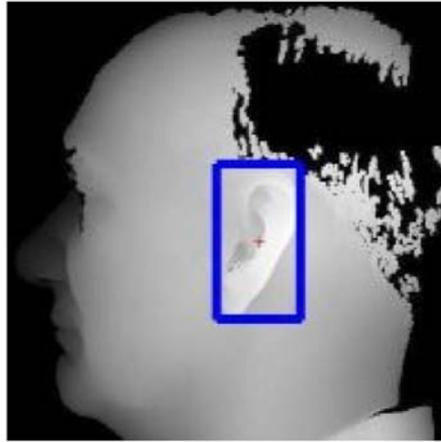


Figura 2.5: Orecchio individuato

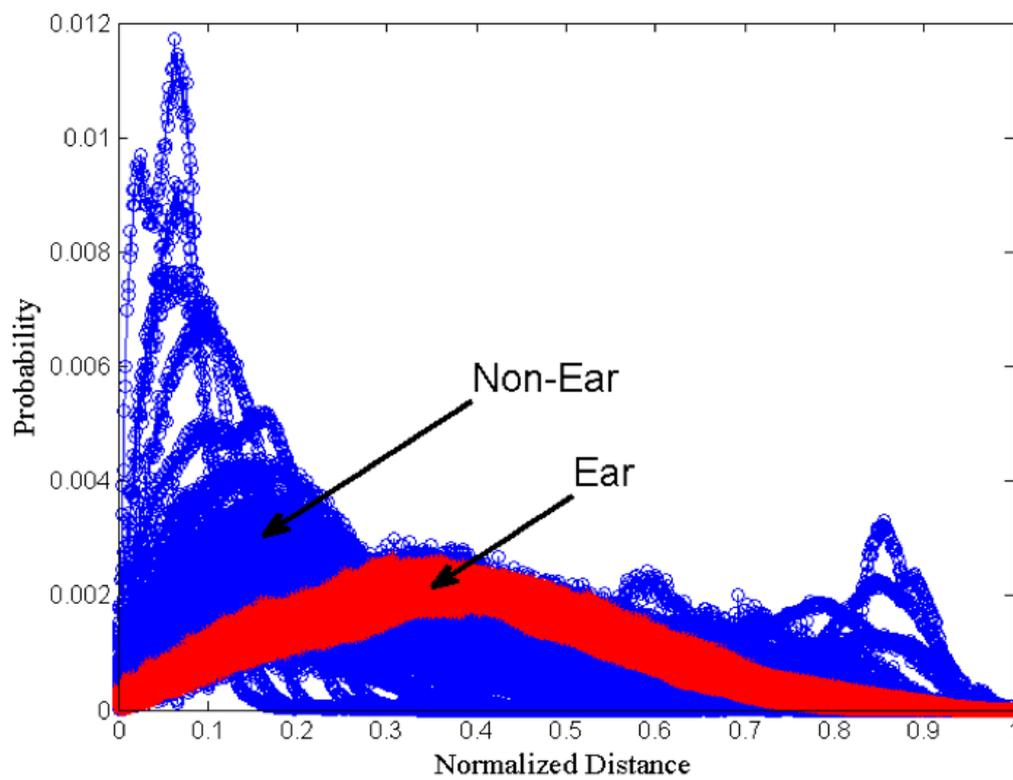


Figura 2.6: Funzione di distribuzione di 200 campioni d'orecchio e 200 campioni di non-orecchio

ning di elementi scelti a caso fra qualche esempio di orecchio 3D. Il modello

d'orecchio è ottenuto calcolando la media delle distribuzioni di forma che si sono ottenute dagli elementi dell'insieme di training. La media è fatta fra i rispettivi valori di ogni distribuzione di forma ottenuta precedentemente.

2.4.2 Identificazione dell'orecchio

Siano le orecchie candidate $I_E = \{I_1, I_2, \dots, I_\eta\}$ con η cardinalità di I_E e I_k è la porzione dell'immagine del profilo 3D che rappresenta il k-esimo candidato dell'orecchio con $k = 1, \dots, \eta$. Per l'identificazione si calcola la distribuzione di forma di ogni candidato. L'identificazione del vero orecchio è ottenuta paragonando il modello d'orecchio con la distribuzione di forma di ogni candidato in I_E . La comparazione fra due distribuzioni f e g è ottenuta attraverso la distanza di Bhattacharyya:

$$D(f, g) = 1 - \sum_{i=1}^n \{\sqrt{f_i g_i}\}$$

dove f_i e g_i sono il numero dei campioni di f e g nell' i -esimo valore e n è il numero totale di valori usati nella distribuzione. Sia $T_E = \{T_{I_1}, T_{I_2}, \dots, T_{I_\eta}\}$ l'insieme delle distribuzioni di forma per le orecchie candidate presenti in I_E . Per ottenere il vero orecchio si calcola la distanza di Bhattacharyya fra il modello T e tutti gli elementi di T_E al fine di ottenere il vettore `matchScore`. Il vero candidato I_ξ ha il minimo valore di distanza nel vettore: $\xi = \arg \min_i \{\text{matchScore}[i]\}$

2.5 Scala e rotazione invarianza

2.5.1 Invarianza alla scala

La tecnica proposta utilizza il grafo di connettività G . Per rendere il criterio di decisione scala invariante, il criterio che connette i vertici in G deve essere scala invariante. Nella tecnica proposta i vertici sono connessi attraverso l'intersezione degli insiemi di inviluppo che non tengono conto della scala dell'immagine. Quindi anche la tecnica proposta è scala invariante.

2.5.2 Invarianza alla rotazione

Dal momento che il criterio che connette i vertici del grafo G uso l'inviluppo convesso, esso è invariante alla rotazione e quindi anche la tecnica proposta lo è.

2.6 Risultati

Gli esperimenti sono stati condotti sulla collezione J2 (UND-J2) [18] del database pubblico dell'università di Notre Dame formato da 1780 immagini di profili 3D di 415 individui. Le immagini da cui non si potevano acquisire le informazioni con lo scanner 3D sono state scartate e sono rimaste 1604 immagini. Tali immagini erano influenzate da variazioni di scala e variazioni del punto di visione. Alcune immagini presentavano occlusione da capelli e orecchini.

2.6.1 Valutazione delle prestazioni

La tabella riassume l'accuratezza della tecnica usata:

Immagini	Accuratezza	Tipologia immagine
1604	99,38%	Immagini 3D normali
1604	99,38%	Immagini ruotate di $+90^\circ$
1604	99,38%	Immagini ruotate di -90°
1604	99,38%	Immagini capovolte orizzontalmente
194	100%	Immagini di varie scale
149	99,32%	Immagini con punto di vista variato

Tabella 2.1: Accuratezza nella registrazione dell'orecchio 3D sul database UND-J2

La rotazione delle immagini è fatta per dimostrare le stesse prestazioni per il riconoscimento dell'orecchio destro e sinistro. Si sono, poi, scelte 194 immagini di differenti scale e 149 con punti di visuale differenti. Nella variazione del punto di vista l'algoritmo fallisce solo nel caso di variazione di angolazione acuta che non consente di vedere le caratteristiche strutturali dell'orecchio. Dalle prestazioni ottenute si osserva che i piccoli punti di rumore derivanti dal processo di scansione 3D delle immagini non intaccano le prestazioni. Le prestazioni peggiorano in presenza di mezzi occlusivi come i capelli.

Capitolo 3

Riconoscimento 2D. An efficient ear recognition technique invariant to illumination and pose

Il paper “An efficient ear recognition technique invariant to illumination and pose” di Surya Prakash e Phalguni Gupta (2011) descrive una tecnica per il **riconoscimento delle immagini 2D** dell’orecchio esterno.

Esso presenta una efficiente tecnica di riconoscimento che sfrutta i benefici delle caratteristiche locali dell’orecchio e cerca di gestire i problemi dovuti all’illuminazione, al poco contrasto, al cambio di illuminazione e a difetti di registrazione. Si usano:

- 3 tecniche di ottimizzazione in parallelo delle immagini per neutralizzare l’effetto del poco contrasto, rumore e illuminazione.
- 1 tecnica di estrazione delle caratteristiche locali (SURF) per migliorare le immagini minimizzando l’effetto delle variazioni di posa e della bassa qualità di registrazione delle immagini.

Si esegue l’estrazione di caratteristiche tramite SURF sulle immagini ottimizzate per ottenere 3 insiemi di caratteristiche locali; un insieme per ogni immagine migliorata. Su questi tre insiemi di caratteristiche si addestrano 3 classificatori. I risultati di matching risultanti dai 3 classificatori sono fusi per la decisione finale. L’algoritmo è valutato su due database: IIT Kanpur ear database e il database ear dell’università di Notre Dame (Collections E).

Al fine di comprendere al meglio il funzionamento dell’algoritmo proposto subito di seguito si andranno ad esplicitare dei concetti teorici dati per acquisiti dall’algoritmo proposto stesso.

3.1 Prerequisiti

3.1.1 Operatore gradiente direzionale

L'operatore descritto in precedenza nella sezione 1.2.4, determina i bordi calcolando prima le componenti ortogonali del gradiente in direzione orizzontale e verticale, e successivamente i gradienti sono stimati sommando in modo appropriato tali componenti. In diverse applicazioni è conveniente calcolare i bordi per un numero definito di direzioni. Questo può essere realizzato mediante la convoluzione dell'immagine di input $f(i, j)$ con diverse maschere h_k i cui pesi modellano le risposte impulsive del gradiente direzionale. Il gradiente direzionale è espresso da

$$g_k(i, j) = f(i, j) * h_k$$

dove h_k è la risposta impulsiva del gradiente nella direzione $\theta_k = \frac{\pi}{2} + k\frac{\pi}{4}$ dove k può assumere valori da 0 (per indicare la direzione nord) a 7. L'operatore gradiente direzionale per ogni pixel è definito come:

$$g(i, j) = \max_k |g_k(i, j)| \quad k = 0, \dots, 7$$

che indica l'inclinazione del bordo nella direzione del massimo valore dei gradienti direzionali calcolati nelle otto direzioni. Le maschere di convoluzione corrispondenti alle k direzioni sono ottenute partendo con $k = 0$ e ruotando in modo circolare di $\pi/4$ ogni elemento esterno della maschera 3×3 . Nella Fig. 3.1 sono rappresentate tutte le maschere di Robinson livello-3 (una derivazione di Sobel).

La somma dei pesi di ogni maschera è zero. Da ciò consegue che nelle zone omogenee dell'immagine l'operatore gradiente direzionale darà sempre come risultato zero. I gradienti direzionali avranno il valore massimo quando la configurazione dei livelli di grigio dei pixel meglio concordano con i modelli rappresentati dalle maschere. Altri operatori gradiente possono essere definiti con rotazioni per esempio di $\pi/6$ con maschere di dimensioni 5×5 . Con maschere ancora più grandi si possono calcolare le direzioni dei bordi in modo più preciso con una buona riduzione del rumore ma con la necessità di notevole tempo di calcolo.

3.1.2 Operatore laplaciano

Per accentuare ulteriormente i bordi di un immagine si può usare un operatore lineare isotropo che può essere definito con la differenziazione del secondo ordine. Si usa l'operatore di Laplace che, per una funzione continua

Direzione Gradiente	Robinson livello-3
Est H ₁	$\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$
NE H ₂	$\begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix}$
Nord H ₃	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$
NW H ₄	$\begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$
Ovest H ₅	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$
SW H ₆	$\begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}$
Sud H ₇	$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$
SE H ₈	$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix}$
Fattore di scala	$\frac{1}{3}$

Figura 3.1: Maschere 3×3 per l'estrazione dei gradienti direzionali

bidimensionale $f(x, y)$, è così definito:

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

che rappresenta la somma delle derivate seconde di una funzione calcolate nella direzione degli assi coordinati x ed y .

L'effetto del laplaciano per una funzione a gradino, (ossia un bordo ideale), può essere rappresentato graficamente come in Fig 3.2.

Si osserva come il bordo può essere rappresentato da più pixel nell'immagine gradiente (in corrispondenza del picco del gradiente). Nella funzione laplaciana il pixel risulta ben localizzato visto che coincidente con il punto dove tale funzione passa per lo zero (da non confondere con l'andamento costante uguale a zero che si verifica quando f è costante). Questo suggerisce

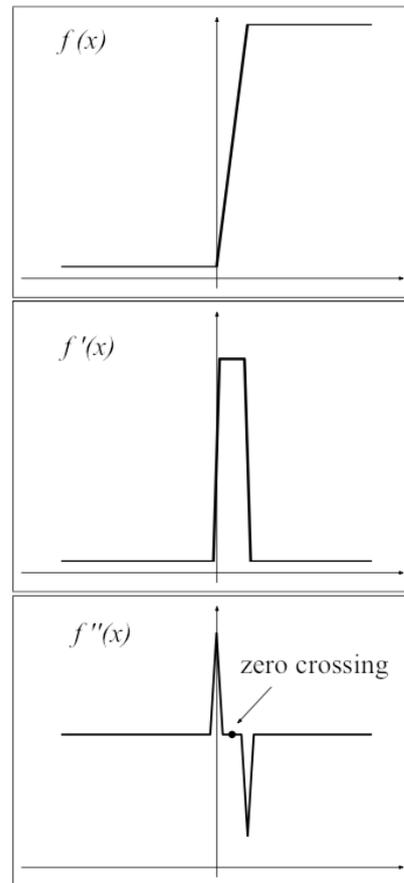


Figura 3.2: Laplaciano di una funzione monodimensionale

l'idea di applicare l'operatore laplaciano ad una immagine $f(i, j)$ per localizzare facilmente i pixel di bordo corrispondenti al pixel dove il fianco di salita o di discesa di $\nabla^2 f$ passa per lo zero (da positivo a negativo o viceversa). Questa transizione di $\nabla^2 f$ per lo zero è chiamata anche "zero crossing" e corrisponde al pixel del bordo. L'operatore laplaciano, basandosi sulla derivata seconda, è più sensibile al rumore rispetto agli altri operatori, non fornisce informazioni di direzione del bordo, e può generare bordi doppi.

Nel dominio spaziale discreto, l'operatore laplaciano può essere approssimato considerando le differenze dei valori di intensità invece delle derivate. La componente orizzontale dell'operatore laplaciano è approssimata come

segue:

$$\begin{aligned}\frac{\partial^2 f}{\partial x^2} &= \frac{\partial f_x}{\partial x} = \frac{\partial(f(i, j+1) - f(i, j))}{\partial x} = \frac{\partial f(i, j+1)}{\partial x} - \frac{\partial f(i, j)}{\partial x} \\ &= (f(i, j+2) - f(i, j+1)) - (f(i, j+1) - f(i, j)) \quad (\text{differenza in avanti}) \\ &= f(i, j+2) - 2f(i, j+1) + f(i, j)\end{aligned}$$

Con la traslazione di un pixel in orizzontale, ottenuta sostituendo j con $j-1$, si calcola il valore approssimato del laplaciano nel punto (i, j) definito da:

$$\frac{\partial^2 f}{\partial x^2} = f(i, j+1) - 2f(i, j) + f(i, j-1)$$

operando in modo analogo per la componente verticale si ottiene:

$$\frac{\partial^2 f}{\partial y^2} = f(i+1, j) - 2f(i, j) + f(i-1, j)$$

Queste approssimazioni conducono alle corrispondenti maschere di convoluzione orizzontale e verticale:

$$h_R = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \quad h_C = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}$$

Dalla combinazione delle componenti orizzontali e verticali si ottiene una singola maschera di convoluzione del laplaciano:

$$h(i, j) = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

3.1.3 Haar wavelet

Le wavelet sono funzioni che soddisfano determinati requisiti. Il nome “wavelet” sta a indicare che esse dovrebbero poter integrare sullo zero “ondeggiano” sopra e sotto l’asse x . Ci sono molti tipi di wavelet; le più semplice sono le Haar wavelet. Haar usò queste funzioni per dare un esempio di un sistema ortonormale numerabile. Come il seno e coseno per l’analisi di Fourier, le wavelet sono usate come le funzioni base per rappresentare altre funzioni. Dal momento che la funzione “wavelet madre” $\psi(x)$ è fissa, si possono formare delle altre wavelet dalla traslazione e dilatazione della wavelet madre

$\{\psi(\frac{x-b}{a}), (a, b) \in R^+ \times R\}$. Nel definire la base delle wavelet è utile usare i seguenti valori $a = 2^{-j}$ e $b = k \cdot 2^{-j}$ con k e j interi. La funzione wavelet madre è:

$$\psi(t) = \begin{cases} 1 & 0 \leq t < 1/2, \\ -1 & 1/2 \leq t < 1, \\ 0 & \text{altrimenti.} \end{cases}$$

Mentre l'analisi di Fourier ha le funzioni di base che sono localizzate sul dominio della frequenza e non nel tempo. Cosicché piccoli cambiamenti di frequenza nella trasformata di Fourier produrranno cambiamenti in qualsiasi possibile punto nello spazio temporale. Le wavelet sono localizzate sia in frequenza (grazie alle dilatazioni) sia nel tempo (grazie alle traslazioni).

Inoltre molte classi di funzioni sono rappresentate in modo più compatto grazie alle wavelet.

Possono essere usate anche per la compressione dei dati visto che il dispendio computazionale è ottimo: la fast wavelet transform ha una complessità computazionale di $O(n)$.

3.1.4 Trasformata Speeded Up Robust Feature (SURF)

SURF è un descrittore e scopritore di punti invarianti alla rotazione. È pensato per estrarre punti caratteristici altamente distintivi ed invarianti. È robusto ai cambiamenti del punto di vista, della scala, dell'illuminazione e dell'occlusione. Identifica i punti salienti detti keypoint. Usa la matrice Hessiana per l'individuazione dei keypoint. Per un dato punto $p(x, y)$ in un immagine I la matrice Hessiana $H(p, \sigma)$ alla scala σ è definita come:

$$H(p, \sigma) = \begin{bmatrix} L_{xx}(p, \sigma) & L_{xy}(p, \sigma) \\ L_{yx}(p, \sigma) & L_{yy}(p, \sigma) \end{bmatrix}$$

$L_{xx}(p, \sigma)$, $L_{xy}(p, \sigma)$, $L_{yx}(p, \sigma)$ e $L_{yy}(p, \sigma)$ sono, rispettivamente, le convoluzioni delle derivate gaussiane del secondo ordine $\frac{\partial^2}{\partial x^2}g(\sigma)$, $\frac{\partial^2}{\partial x \partial y}g(\sigma)$, $\frac{\partial^2}{\partial y \partial x}g(\sigma)$ e $\frac{\partial^2}{\partial y^2}g(\sigma)$ con l'immagine I nel punto p . Per velocizzare la computazione, le derivate gaussiane del secondo ordine nella matrice Hessiana sono approssimate usando dei filtri "box" (che evidenziano l'interno di regioni rettangolari). Per individuare i keypoint a differenti scale, la rappresentazione dello spazio scalato dell'immagine è ottenuto facendo la convoluzione dell'immagine coi filtri "box". Lo spazio di scala è analizzato aumentando la regione occupata del filtro piuttosto che riducendo iterativamente la dimensione dell'immagine. Per localizzare i punti di interesse nell'immagine e sulle zone scalate si

implementa la soppressione dei non-massimi (come introdotto nell'algoritmo 2.1.2) in un vicinato 3×3 .

Per generare un vettore descrittore del keypoint si considera una regione circolare attorno al keypoint e si calcolano, nelle direzioni verticali ed orizzontali, le risposte Haar wavelet dx e dy . Queste risposte sono usate per calcolare la direzione dominante nella regione circolare. I vettori caratteristici sono valutati in funzione della direzione dominante e quindi risultano invarianti alla rotazione d'immagine. Inoltre si considera una regione quadrata attorno ad ogni keypoint che viene allineata con la direzione dominante. La regione quadrata è divisa 4×4 sotto-regioni e la risposta alle Haar wavelet è calcolata per ogni sotto-regione. Le somme delle risposte wavelet nelle direzioni orizzontali e verticali per ogni sotto-regione sono usate come valori caratteristici. Inoltre il valore assoluto delle risposte è sommato per ottenere informazioni sulla polarità dei cambiamenti d'intensità dell'immagine. Cosicché il vettore caratteristico \vec{v}_i per i -esima sotto-regione è dato da:

$$\vec{v}_i = \left\{ \sum dx, \sum dy, \sum |dx|, \sum |dy|, \right\}$$

Il vettore caratteristico SURF di un keypoint è ottenuto concatenando i vettori caratteristici \vec{v}_i da tutte le 16 sotto-regioni attorno al keypoint; si ottiene come risultato un vettore da 64 elementi. La versione estesa del SURF (detta SURF-128), che è più distintiva, aggiunge coppie di caratteristiche distintive. Si utilizza sempre la somma come descritta precedentemente, ma si dividono i risultati ulteriormente. Le somme dx e $|dx|$ sono calcolate separatamente per $dx < 0$ e $dy \geq 0$; similmente le somme di dy e $|dy|$ sono calcolate in accordo col segno di dx , perciò si raddoppia il numero delle caratteristiche. L'algoritmo per il riconoscimento dell'orecchio usa la tecnica SURF-128 (indicata d'ora in poi con SURF) per la rappresentazione delle caratteristiche.

Il matching nel SURF è ottenuto accoppiando i rapporti tra confinanti più simili. Il candidato migliore al matching per un keypoint in un'immagine è il confinante più vicino tra i keypoint nell'immagine di test; i confinanti più vicini sono i keypoint con distanza euclidea minima dal dato vettore descrittore. La probabilità che il match sia corretto è determinata calcolando il rapporto di distanza fra il più vicino confinante e la distanza del secondo più vicino confinante. Tutti i match in cui il rapporto di distanza è maggiore della soglia τ sono scartati.

3.1.5 Equalizzazione adattiva a istogramma

L'equalizzazione adattiva ad istogramma (Adaptive histogram equalization ADHist) può essere usata per migliorare il contrasto dell'immagine. Essa

divide l'immagine in "mattonelle" multiple non sovrapposte e calcola l'equalizzazione ad istogramma¹ per ognuna di esse. Questo aumenta il contrasto di ogni mattonella. Le mattonelle confinanti sono combinate assieme per ottenere l'intera immagine migliorata. ADHist usa l'interpolazione bilineare per rimuovere bordi artificialmente indotti durante la combinazione. ADHist consente di aumentare il contrasto locale di un'immagine facendo emergere più dettagli nell'immagine. Sia I l'immagine di dimensioni $a \times b$ che si deve migliorare. Essa è divisa in mattonelle di dimensione $\alpha \times \beta$ dove $\alpha < a$ e $\beta < b$. Le mattonelle sono migliorate individualmente e cucite assieme per avere l'immagine complessiva migliorata. La scelta di α e β ha effetto sulle prestazioni del miglioramento. Nel paper saranno scelte empiricamente. In Fig 3.3 vi è un esempio.

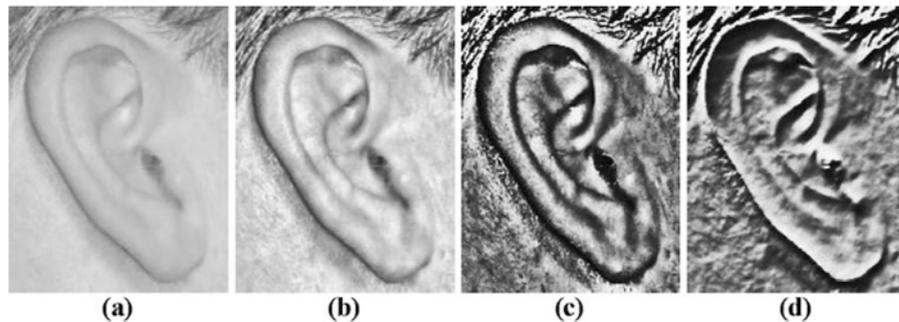


Figura 3.3: Esempio di miglioramento dell'immagine. (a) Immagine originale. (b)ADHist. (c) NLM. (d) SF.

3.1.6 Filtro a mezzi non locali

L'algoritmo del filtro a mezzi non locali (non-local means NLM) è proposto per migliorare l'immagine rimuovendone i difetti. Esso considera i valori dei pixel dell'intera immagine al fine di ridurre il rumore nella medesima. L'algoritmo utilizza il fatto che per ogni piccola finestra di un'immagine esistono diverse finestre simili nell'immagine e tutte queste finestre possono essere sfruttate per ridurre il rumore dell'immagine. Sia la rumorosità dell'immagine di dimensione $a \times b$ indicata con $I_n(p)$ dove p sta per la locazione di un pixel arbitrario in I_n . L'algoritmo NLM ottiene il valore di ogni pixel dell'immagine a cui è stato ridotto il rumore $I_a(p)$ come la media pesata dei

¹L'equalizzazione ad istogramma prevede di equalizzare la funzione (livelli di grigio, numero pixel per livello) associata ad un'immagine. Un'immagine col giusto contrasto occuperà l'ascissa dei livelli di grigio in modo equilibrato e ne enfatizzerà il contrasto

pixel di $I_n(p)$ come segue:

$$I_d(p) = \sum_{p \in I_n(p)} \rho(z, p) I_n(p)$$

dove $\rho(z, p)$ è la funzione di ponderazione che calcola la similarità tra il vicinato alla locazione del pixel z e p . Essa è definita come segue:

$$\rho(z, p) = \frac{1}{\eta(z)} \exp\left(-\frac{G_\sigma \|I_n(\Omega_p) - I_n(\Omega_z)\|_2^2}{h^2}\right)$$

dove

$$\eta(z) = \sum_{p \in I_n(p)} \exp\left(-\frac{G_\sigma \|I_n(\Omega_p) - I_n(\Omega_z)\|_2^2}{h^2}\right)$$

e dove G_σ è la funzione kernel gaussiano² con deviazione standard σ , Ω_p e Ω_z sono i vicinati locali alla locazione dei pixel p e z rispettivamente, h è il parametro che controlla il decadimento della funzione esponenziale ed $\eta(z)$ rappresenta il fattore di normalizzazione. Da queste equazioni si può osservare che se il vicinato locale della posizione di una coppia di pixel ha un alto grado di similarità, a tali punti sarà assegnato un alto peso quando sarà da valutare il loro valore di diminuzione del rumore.

Una selezione appropriata N della dimensione del vicinato e del parametro di decadimento h consente di ottenere un'immagine attenuata ma coi contorni preservati. Un esempio di immagine a cui è stato diminuito il rumore usando l'NLM con una dimensione 3×3 pixel del vicinato locale e $h = 50$ è mostrata in Fig 3.3.

3.1.7 Filtro orientabile

Il filtro orientabile (steerable filter SF) fornisce un modo efficiente per sintetizzare i filtri di orientazione arbitraria attraverso la combinazione lineare di un insieme di filtri di base. Ciò consente all'utente di "manovrare" in modo adattivo un filtro su ogni direzione e di determinare l'output del filtro come una funzione dell'orientamento. Per esempio per definire i filtri orientabili si può usare la funzione gaussiana come base del filtro. I filtri orientabili sono applicati in molte elaborazioni d'immagini e nella computer vision come filtri di adattamento dell'angolazione, miglioramento dell'immagine, ombreggiatura delle forme, rilevazione dei bordi... Essi si sono dimostrati efficaci anche nella rappresentazione dell'illuminazione invariante di un immagine. Per ottenere la rappresentazione dell'illuminazione invariante si possono usare filtri

²La funzione kernel gaussiano in 2D è definita come $G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$

derivati da filtri gaussiani orientabili su multiple scale ed orientazioni. L'immagine filtrata può essere calcolata dalla combinazione lineare pesata delle immagini filtrate ottenute dopo l'applicazione di questi filtri all'immagine di input.

I filtri orientabili basati sulla funzione gaussiana sono definiti da due parametri: σ che definisce la scala del filtro e θ che definisce l'orientamento del filtro. Per definire le funzioni gaussiane su scale multiple si usa un insieme di valori per σ e θ ; siano tali valori $\{\sigma_1, \sigma_2, \dots, \sigma_l\}$ e $\{\theta_1, \theta_2, \dots, \theta_n\}$ dove ogni coppia (σ_i, θ_j) definisce una base del filtro. La spaziatura angolare per definire i filtri è comunemente scelta uguale; perciò n valori ugualmente disegnati fra 0° e 180° definiscono n valori di θ . Le scelte dei valori di σ e n dipendono rispettivamente dalla dimensione e dal contenuto dell'immagine. Un esempio del miglioramento dell'immagine è dato in Fig 3.3; si ottiene un'immagine normalizzata all'illuminazione.

3.2 Algoritmo proposto

Il sistema di riconoscimento dell'orecchio segue 3 passi principali: miglioramento dell'immagine, estrazione della caratteristica e classificazione e fusione. Una visione d'insieme del sistema è data dalla Fig. 3.4

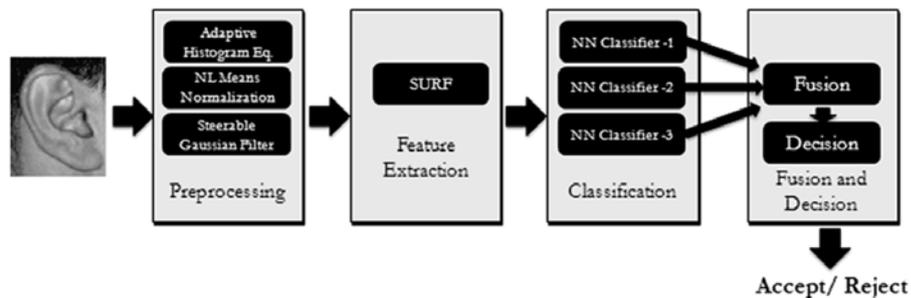


Figura 3.4: Diagramma a blocchi dell'algoritmo proposto

3.2.1 Miglioramento delle immagini

Questo passo racchiude 3 tecniche di miglioramento dell'immagine ed è progettato per migliorare il contrasto dell'immagine dell'orecchio e per normalizzare l'effetto dell'illuminazione e dell'ombreggiamento. Lo scopo del miglioramento è ottenere i corretti vettori descrittivi SURF per i punti caratteristici e aiutare a stabilire la corretta corrispondenza tra i punti caratteristici in due

immagini. Per esempio un particolare punto caratteristico in due immagini differenti dello stesso oggetto (che sono diversamente illuminate) possono ottenere 2 diversi descrittori dei vettori SURF in assenza di miglioramento. Ma quando è applicato il miglioramento i vettori caratteristici di punti corrispondenti in due immagini sono molto simili.

L'algoritmo sul miglioramento, che prevede le sezioni 3.1.5, 3.1.6 e 3.1.7, è usato in parallelo su tutte le immagini dell'orecchio di input, al fine di ottenere l'immagine migliorata che poi sarà usata per l'estrazione delle caratteristiche.

3.2.2 Estrazione delle caratteristiche

Questo passo usa la tecnica SURF per l'estrazione delle caratteristiche che fornisce la rappresentazione dell'immagine in termini di un insieme di punti caratteristici; ad ogni punto è associato un vettore descrittivo di 128 elementi caratteristici. Le caratteristiche SURF sono efficientemente capaci di catturare le proprietà di localizzazione spaziale, cambiamento nel punto di vista 3D, orientamento e sensibilità di scala. Esso fornisce caratteristiche altamente distintive nel senso che una singola caratteristica può essere correttamente matchata con alta probabilità in un ampio database di caratteristiche ottenute da molte immagini.

Si propone una tecnica per il livello di fusione caratteristico perché si vuole ottenere il modello rappresentativo fuso di un soggetto, combinando le caratteristiche di multipli campioni di studio del soggetto stesso. Se n sono i campioni del soggetto forniti per l'addestramento, si ottiene un modello rappresentativo caratteristico per il soggetto fondendo i punti caratteristici assieme e considerando i punti caratteristici ridondanti solo una volta. Siano n i modelli di caratteristiche biometriche, di un soggetto usato come addestramento, rappresentati da F_1, F_2, \dots, F_n . Un modello biometrico di caratteristiche fuse F_{fused} è ottenuto come segue:

$$F_{fused} = F_1 \cup F_2 \cup \dots \cup F_n$$

dove la cardinalità dell'insieme F_{fused} fornisce il numero di punti caratteristici presenti nel modello fuso. La fusione dei modelli è fatta in modo incrementale dove i primi 2 modelli caratteristici F_1 e F_2 sono fusi per generare un nuovo modello T il quale è fuso col modello caratteristico F_3 . Tale procedura continua finché tutti i modelli caratteristici sono fusi assieme. Mentre si fondono due modelli caratteristici F_i e F_{i+1} si applica il SURF matching tra i modelli per trovare i punti ridondanti. Se un punto caratteristico in un modello combacia con un punto caratteristico in un altro modello è considerato comune da entrambi ed è usato solo una volta nella fusione.

3.2.3 Classificazione e fusione

Le caratteristiche estratte, ottenute da ogni immagine migliorata, sono usate per addestrare il classificatore confinante più vicino. La strategia di matching nel classificatore confinante più vicino è la seguente: un punto di interesse nell'immagine di test è comparato ad un punto di interesse nel modello di riferimento calcolando la distanza euclidea tra i loro vettori descrittivi. Il matching di due punti caratteristici è ottenuto considerando il rapporto di strategia di matching confinante più vicino; una coppia di matching è individuata se la sua distanza è minore di τ volte la distanza del secondo confinante più vicino (con τ soglia di match).

Il risultato di matching fra due immagini d'orecchio è ottenuto sulla base del numero di punti caratteristici accoppiati tra le due immagini. Questi risultati di matching sono normalizzati usando la normalizzazione min-max e sono fusi usando la regola della somma pesata. La decisione di classificazione finale è presa usando il risultato della fusione.

3.3 Risultati degli esperimenti

I risultati di un sistema biometrico possono essere misurati in termini di accuratezza del riconoscimento, rapporto d'errore uguale (equal error rate EER) ed errore sotto le curve ROC (error under ROC curves EUC). L'accuratezza del riconoscimento è utilizzata nei sistemi di verifica ed è definita come $100 - \frac{(FAR+FRR)}{2}$ dove FAR (False Acceptable Rate) indica il rapporto in cui si accetta un impostore come la persona corretta e FRR (False Rejection Rate) è il rapporto nel quale una persona corretta è ingiustamente scartata come impostore. EER è definito come il rapporto in cui FAR e FRR sono uguali. Le prestazioni del sistema di verifica possono anche essere valutate usando la curva dell'operatore ricevitore caratteristico (ROC), la quale, graficamente, dimostra i cambiamenti di GAR (Genuine Acceptable Rate definito come $100 - FRR$) con cambiamenti in FAR.

3.3.1 Database

Gli esperimenti si sono svolti su 2 database: il database "IIT Kanpur" e il database dell'università di Notre Dame (collezione E).

Il primo database è composto da due data set. Il data set 1 contiene 801 immagini di profili di facce raccolte da 190 soggetti. Il numero di facce associate ad un singolo individuo varia da 2 a 10. Un esempio delle immagini del data set 1 è nella Fig. 3.5 (a). Il data set 2 consiste di 801 immagini di profili di facce raccolte da 89 individui. Per ogni soggetto sono catturate 9

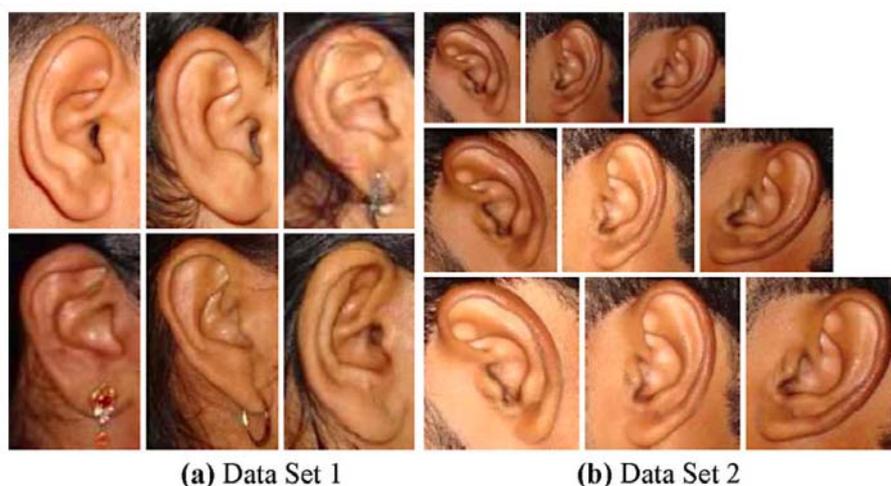


Figura 3.5: Esempi del database IIT Kanpur

immagini: di ogni immagine vi sono 3 rotazioni e vi sono 3 scale differenti per ogni rotazione. Le immagini del data set 2 consistono di un punto di vista frontale dell'orecchio catturate in 3 posizioni diverse: quando la persona guarda dritta, quando la persona guarda circa 20° in basso e quando guarda circa 20° in alto. In tutte queste posizioni l'immagine è catturata con 3 differenti scale, posizionando la fotocamera a circa $1m$ e settando lo zoom della telecamera a 1,7x, 2,7x e 3,3x. La Fig. 3.5 (b) mostra 9 immagini dal data set 2 di un individuo. Lo scopo nell'uso di multiple tipologie di dato è dimostrare la robustezza dell'approccio proposto.

Il database dell'università di Notre Dame (collezione E, abbreviato in UND-E) consiste di 464 immagini del profilo di un volto di 114 soggetti; da 3 a 9 immagini appartengono ad un singolo soggetto. Le immagini sono raccolte in giorni differenti con differenti condizioni di posa e illuminazione. Alcuni esempi di immagini ottenute dal database sono riportate in Fig. 3.6. si può notare che vi è una notevole variazione fra le immagini della stessa persona dovuta alla variazione di posizione e a differenti condizioni d'immagine.

Entrambi i database contengono immagini di profili di soggetti umani. Le orecchie sono divise dal profilo dell'immagine del soggetto usando una tecnica di segmentazione discussa in [24]. Si esegue una divisione manuale per le immagini (4%) dove la tecnica [24] è deficitaria.

3.3.2 Parametri

La selezione dei valori appropriati dei parametri è critica per ottenere le performance migliori dalla tecnica proposta. I parametri che hanno grande



Figura 3.6: Esempi del database dell'università di Notre Dame Collection E

impatto sulle prestazioni sono: la dimensione delle “mattonelle” nel ADHist, i valori di σ e n nel SF, i valori di h e N nel NLM e il valore di τ nel matching SURF.

Dal momento che risulta difficile testare la tecnica proposta per ogni possibile valore dei 6 parametri, i parametri sono settati ai loro valori ottimali euristicamente. Per ottenere i valori ottimali un insieme di 25 soggetti è selezionato casualmente da ogni database e il settaggio dei parametri è eseguito solo su tale insieme. Questi parametri sono poi usati per testare l'intero database.

Dimensione delle mattonelle per ADHist

La tecnica proposta considera la dimensione quadrata delle mattonelle nell'approccio ADHist (quindi $\alpha = \beta$). Le dimensioni della mattonella variano da 2×2 a 20×20 e, per ogni dimensione, si calcolano EER e EUC del sistema solo quando viene usata ADHist come tecnica per il miglioramento dell'immagine. La dimensione della mattonella che fa ottenere il minimo EER è scelta come dimensione ottimale. Inoltre se con due dimensioni della mattonella si ottiene lo stesso EER, si utilizza il valore della mattonella col corrispondente valore EUC minore come valore ottimo. Dagli esperimenti il valore ottimo delle mattonelle per il database IITK set 1 e set 2 è 8×8 e 4×4 rispettivamente mentre per il database UND-E è 16×16 .

Si nota che i cambiamenti del EER e EUC sono gradualmente eccetto in alcune eccezioni. Per $\tau = 0,6$ e $\tau = 0,7$ EER e EUC decrescono gradualmente mentre per $\tau = 0,4$ e $\tau = 0,5$ EER e EUC restano costanti. Inoltre per $\tau = 0,3$ il loro comportamento è un po' brusco; il SURF matching a basse soglie non è molto stabile. Si può notare che il data set UND-E ha delle variazioni di contrasto ed illuminazione quindi l'osservazione è leggermente brusca. Ciò non è il caso del data set IITK.

Dagli esperimenti si vede che i valori d'errore sono sempre gli stessi per $\tau = 0, 7$ e per differenti dimensioni di mattonelle comprese tra 2×2 e 10×10 o tra 12×12 e 14×14 o tra 16×16 e 20×20 . Quindi piccoli cambiamenti nella dimensione delle mattonelle non incidono sui valori d'errore.

Valori di h e N nei filtri NLM

Nei filtri NLM h è uno scalare che controlla il decrescere della funzione esponenziale e N è uno scalare che definisce la dimensione del vicinato (ad es. la dimensione dei ritagli da usare nell'algoritmo NLM). Per cercare il valori ottimi h ed N sono fatti variare fra 20 e 100 e fra 4 e 6 rispettivamente. Per ogni combinazione di (h, N) è eseguito il miglioramento dell'immagine e l'immagine migliorata è usata per il riconoscimento. I valori di EER e EUC del sistema sono calcolati e il valore (h, N) che fa corrispondere il minimo EER è considerato come ottimo. Si considera l'EUC nel caso in cui due o più coppie (h, N) diano lo stesso EER. Gli esperimenti condotti per trovare l'EER e l'EUC per i database IITK e UND-E dimostrano che i valori ottimali di (h, N) per il database IITK al data set 1 e data set 2 sono $(100, 6)$ e $(50, 6)$ rispettivamente mentre per il database UND-E sono $(50, 6)$.

Valore di σ e n nel SF

Nei filtri orientabili σ definisce un vettore lungo l , dove l è il numero è il numero delle scale del filtro e n è a risoluzione angolare del filtro. Negli esperimenti si sono considerati due insiemi di σ : $\{0.5, 1, 1.5, 2, 2.5\}$ e $\{0.1, 1, 2, 3, 4\}$ mentre si è scelto n come 4 (quindi $\theta = 0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$), 6 (quindi $\theta = 0, \frac{\pi}{6}, \frac{\pi}{3}, \frac{\pi}{2}, \frac{2\pi}{3}, \frac{5\pi}{6}$) e 8 (quindi $\theta = 0, \frac{\pi}{8}, \frac{\pi}{4}, \frac{3\pi}{8}, \frac{\pi}{2}, \frac{5\pi}{8}, \frac{3\pi}{4}, \frac{7\pi}{8}$). Per ogni combinazione di σ e n si esegue il miglioramento dell'immagine e l'immagine migliorata è usata per il riconoscimento usando le caratteristiche del SURF e il classificatore del vicinato più vicino. Si calcolano i valori di EER e EUC del sistema e il valore (σ, n) associato al minimo EER è considerato come ottimale. Gli esperimenti condotti per trovare EER e EUC per i database IITK e UND-E dimostrano che i valori ottimi dei parametri (σ, n) per SF sono $(\{0.5, 1, 1.5, 2, 2.5\}, 8)$ e $(\{0.1, 1, 2, 3, 4\}, 8)$ per il database IITK data set 1 e data set 2 rispettivamente, mentre $(\{0.1, 1, 2, 3, 4\}, 8)$ per il database UND-E. Inoltre ci sono due valori di coppia (σ, n) ($(\{0.1, 1, 2, 3, 4\}, 6)$ e $(\{0.1, 1, 2, 3, 4\}, 8)$) per i quali il valore EER raggiunge il valore minimo. Cosicché si è scelto $(\{0.1, 1, 2, 3, 4\}, 8)$ come parametro ottimo visto che ad esso era associato il minimo EUC.

Valore di θ per il matching SURF

La correttezza della corrispondenza nel SURF matching è determinata calcolando il rapporto della distanza fra il vicinato più vicino e la distanza del secondo vicinato più vicino. Tutti i match in cui il rapporto di distanza è maggiore di τ sono scartati. Gli esperimenti condotti sui database IITK e UND-E sono stati fatti facendo variare il valore di τ da 0,3 a 0,7 con passo di 0,1. I valori in tale intervallo sono stati usati per in ogni esperimento per determinare i parametri dei filtri ADHlist, NLM e SF.

3.3.3 Risultati

Tab 3.1 e 3.2 forniscono i risultati dell'accuratezza di riconoscimento, (col corrispondente FAR e FRR), EER, EUC per il database IITK per varie combinazioni di tecniche di miglioramento. Si può osservare che i migliori risultati sono ottenuti quando tutte le 3 tecniche di miglioramento sono impiegate nel processo di riconoscimento. Le curve di ROC per il data set 1 e data set 2 sono mostrate in Fig. 3.7 (a) e (b) rispettivamente. Le curve ROC ottenute, per la tecnica che impiega tutte e 3 le tecniche di miglioramento dell'immagine, hanno evidenziato le migliori prestazioni rispetto alle altre.

L'accuratezza ottenuta in Tab. 3.2 è sempre maggiore di quella mostrata in Tab. 3.1 eccetto che per NLM. L'accuratezza maggiore è ottenuta come conseguenza del fatto che nel data set 2 tutti i soggetti hanno 9 immagini associate, mentre nel data set 1 il numero di immagini del soggetto varia da 2 a 10 (il 50% dei soggetti ha meno di 4 immagini associate). Quindi il training set data set 2 è migliore del data set 1 visto che fornisce una maggiore accuratezza.

La Tab. 3.3 fornisce vari valori delle misurazioni di prestazioni per il database UND-E per varie combinazioni di tecniche di miglioramento. Sempre per questo database si nota che i migliori risultati sono ottenuti quando tutte e 3 le tecniche di miglioramento sono impiegate nel processo di riconoscimento. Dalla Tab. 3.3 si osserva che i migliori EER e EUC sono minori di quelli riportati nelle 2 tecniche di riconoscimento ben note [25] e [26]. Una comparazione delle tecniche note con le tecniche proposte risultanti dal database UND-E sono riassunte nella Tab. 3.4. I risultati ottenuti dalla tecnica proposta sono mediati su 30 esperimenti; si dimostra che si ottengono performance più stabili di quelle ottenute in [25] e [26] dove i risultati sono mediati su 10 e 20 esperimenti rispettivamente. Le curve ROC per il database UND-E sono mostrate in Fig. 3.8; la curva ROC che impiega tutte e 3 le tecniche di miglioramento dell'immagine è superiore alle altre.

Il risultato del livello di fusione è ottenuto usando la regola della somma pesata [28]. Si assume che l'uso indipendente dei classificatori C_1, C_2 e C_3 per la classificazione produce un'accuratezza di classificazione A_1, A_2 e A_3 rispettivamente. Nella tecnica proposta queste accuratèzze sono usate per pesare i risultati dei classificatori individuali per la fusione. Il risultato della fusione è dato come segue:

$$S = \frac{A_1 S_1 + A_2 S_2 + A_3 S_3}{A_1 + A_2 + A_3}$$

dove S_1, S_2, S_3 sono i risultati individuali prodotti dai classificatori C_1, C_2 e C_3 rispettivamente. Le curve ROC mostrate nelle Fig. 3.7 e 3.8 sono disegnate per il sistema che usa la regola della somma pesata nella fusione dei risultati di matching ottenuti dai 3 classificatori.

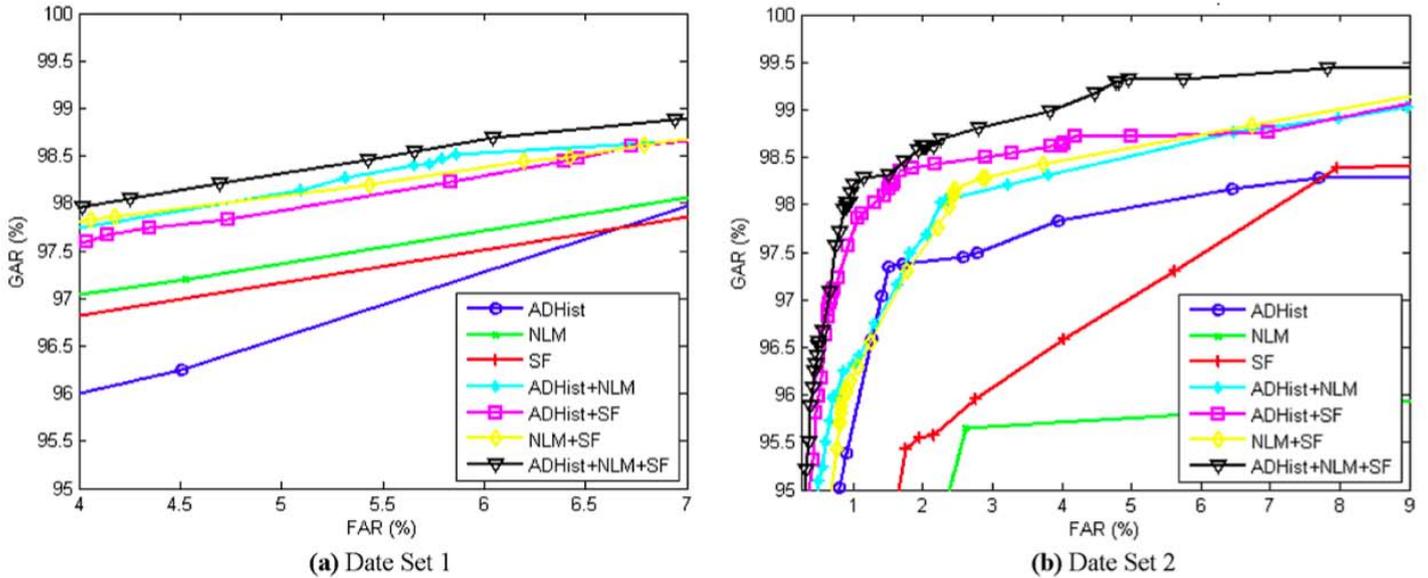


Figura 3.7: Curve ROC per i data set del database IITK con varie combinazioni di tecniche di miglioramento dell'immagine

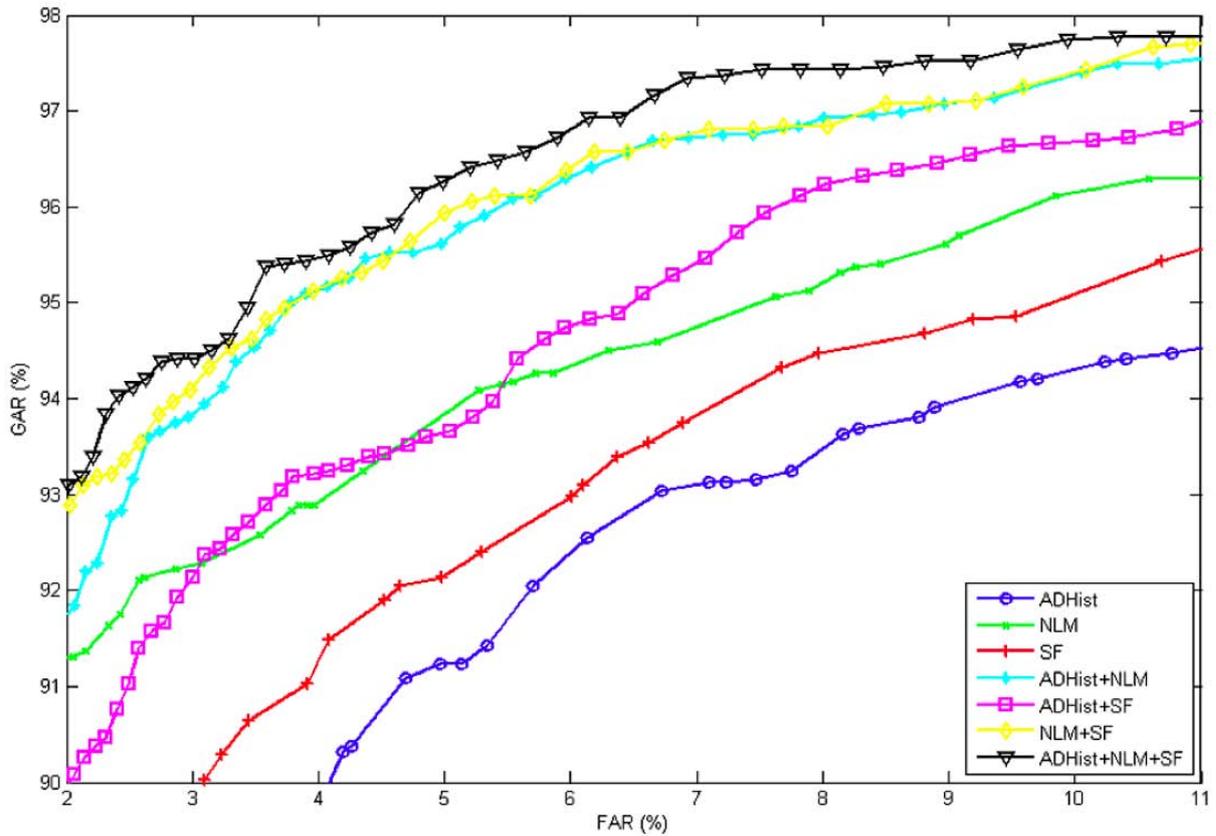


Figura 3.8: Curve ROC per UND-E con varie combinazioni di tecniche di miglioramento dell'immagine

Schema di fusione	Accuratezza (FAR FRR)	EER	EUC
ADHist	96,54 (2,89 4,04)	3,46	1,42
NLM	97,10 (3,07 2,72)	2,90	1,17
SF	96,68 (2,92 3,72)	3,46	0,98
ADHist e NLM	97,25 (2,83 2,67)	2,98	0,90
ADHist e SF	97,13 (2,92 2,82)	3,09	0,80
NLM e SF	97,20 (2,71 2,89)	2,94	0,83
ADHist e NLM e SF	97,35 (2,70 2,60)	2,88	0,75

Tabella 3.1: Prestazioni ottenute sul database IITK data set 1 per varie combinazioni di tecniche di miglioramento. In grassetto i risultati migliori.

Schema di fusione	Accuratezza (FAR FRR)	EER	EUC
ADHist	97,94 (1,42 2,70)	2,25	1,03
NLM	96,55 (2,10 4,79)	3,48	2,32
SF	96,85 (1,70 4,61)	3,28	1,11
ADHist e NLM	98,17 (1,49 2,17)	2,11	0,58
ADHist e SF	98,62 (1,07 1,69)	1,68	0,40
NLM e SF	98,07 (1,83 2,02)	2,26	0,48
ADHist e NLM e SF	98,79 (0,88 1,54)	1,59	0,36

Tabella 3.2: Prestazioni ottenute sul database IITK data set 2 per varie combinazioni di tecniche di miglioramento. In grassetto i risultati migliori.

Schema di fusione	Accuratezza (FAR FRR)	EER	EUC
ADHist	93,64 (5,18 7,54)	6,72	2,40
NLM	95,25 (2,31 7,19)	5,75	1,40
SF	94,17 (3,31 8,36)	6,51	1,67
ADHist e NLM	96,13 (2,97 4,77)	4,40	1,34
ADHist e SF	95,41 (4,01 5,18)	5,06	1,49
NLM e SF	96,31 (2,85 4,53)	4,22	1,13
ADHist e NLM e SF	96,75 (2,58 3,92)	3,80	1,16

Tabella 3.3: Prestazioni ottenute sul database UND-E per varie combinazioni di tecniche di miglioramento. In grassetto i risultati migliori.

Schema di fusione	Accuratezza (FAR FRR)	EER	EUC
Tecnica in [25]	-	4,20	3,00
Tecnica in [26]	-	-	1,50
ADHist e NLM e SF	96,75 (2,58 3,92)	3,80	1,16

Tabella 3.4: Prestazioni comparate con le tecniche note [25] e [26] svolte sul database UND-E. In grassetto i risultati migliori.

Capitolo 4

Riconoscimento 3D. A computationally efficient approach to 3D ear recognition employing local and holistic features

Nel paper “A computationally efficient approach to 3D ear recognition employing local and holistic features ” di Jindan Zhou, Cadavid, S. e Abdel-Mottaleb, M. (2011) si descrive una tecnica per il **riconoscimento delle immagini 3D** dell’orecchio esterno.

In questo metodo si presenta un sistema per il riconoscimento di oggetti 3D capace di distinguere fra orecchi esterni 3D molto simili. Il sistema è formato da 4 parti principali:

- segmentazione 3D dell’orecchio
- estrazione e match delle caratteristiche locali
- estrazione e match delle caratteristiche olistiche¹

¹L’olismo si basa sull’idea che le proprietà di un sistema non possano essere spiegate esclusivamente tramite le sue componenti. Relativamente a ciò che può essere chiamato “olistico”, per definizione, la sommatoria funzionale delle parti è sempre maggiore/differente della somma delle prestazioni delle parti prese singolarmente. Un tipico esempio di struttura olistica è l’organismo biologico, perché un essere vivente, in quanto tale, va considerato sempre come un’unità-totalità non esprimibile con l’insieme delle parti che lo costituiscono.

- piattaforma di fusione delle caratteristiche locali ed olistiche al livello di matching dei risultati.

Per la segmentazione dell'orecchio 3D si utilizza il metodo presente nel [20]. Per l'estrazione delle caratteristiche locali e della loro rappresentazione si usa il descrittore "Histogram of Indexed Shapes" (HIS) e si estende al descrittore per la forma di oggetti centrati 3D detto "Surface Path Istogram of Indexed Shapes" (SPHIS) sia per rappresentare parti di superficie, che per effettuare il matching fra parti di superfici. Per l'estrazione ed il matching di caratteristiche olistiche si propone il "voxeling" della superficie dell'orecchio dopo il quale si può fare un confronto efficace fra coppie di immagini da confrontare. Il risultato della fusione nel matching delle componenti locali e nel match delle componenti olistiche produce il matching finale. Una visione generale dell'algoritmo è data dalla figura 4.1

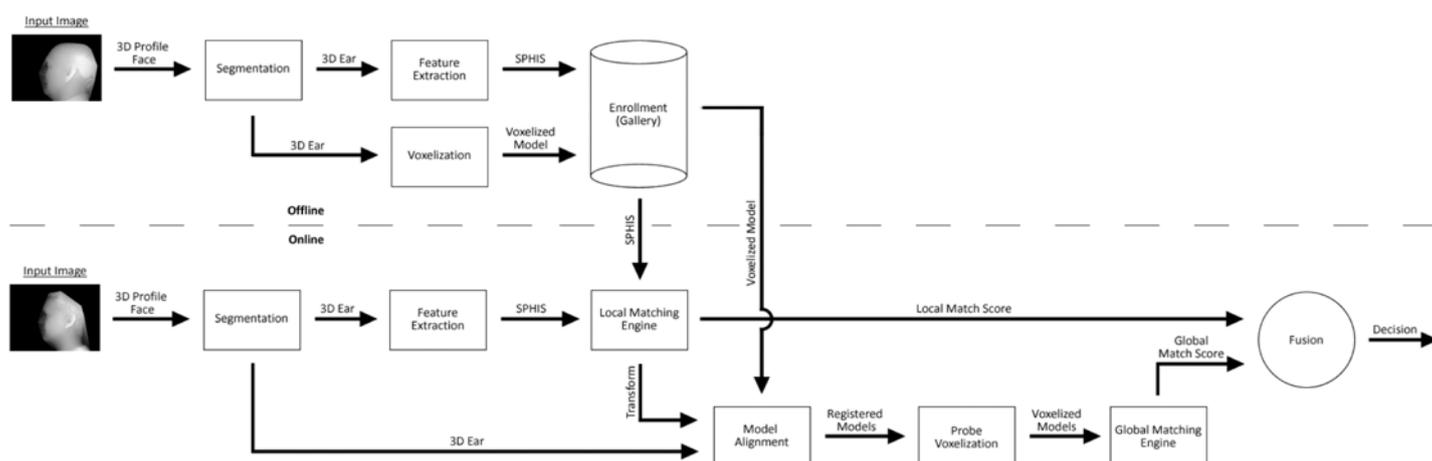


Figura 4.1: Visione d'insieme del metodo

Al fine di comprendere al meglio il funzionamento dell'algoritmo proposto subito di seguito si andranno ad esplicitare dei concetti teorici dati per acquisiti dall'algoritmo proposto stesso.

4.1 Prerequisiti

4.1.1 Matrice di covarianza

La matrice di covarianza rappresenta la variazione di coppie di variabili aleatorie, in una analisi statistica di un fenomeno; rappresenta cioè il modo in cui ogni variabile varia rispetto alle altre.

Sia X l'insieme dei campioni osservati, ciascuno dei quali corrisponde a un vettore di valori. Ciascuno di questi vettori ha lunghezza D . La matrice delle covarianze avrà perciò dimensioni $D \times D$. I suoi valori sono definiti come:

$$\sigma_{ij} = \frac{1}{n} \sum_{h=1}^n (x_{hi} - \mu_j)^2$$

Dove ogni x_{hi} rappresenta una caratteristica (il vettore \vec{x} rappresenta una rilevazione), il vettore $\vec{\mu}$ la media dei valori di ciascuna caratteristica (perciò μ_j rappresenta la j -esima caratteristica), e n è il numero di rilevazioni.

4.1.2 Autofacce

Nella elaborazione digitale delle immagini, le immagini di facce possono essere viste come vettori le cui componenti sono la luminosità dei singoli pixel. La dimensione dello spazio vettoriale in cui sono ambientati è pari al numero di pixel, e gli autovettori di una particolare matrice, detta matrice di covarianza, sono chiamati “autofacce”.

Il metodo di riconoscimento facciale basato sulle “autofacce” è un approccio olistico che sfrutta l'analisi delle componenti principali. Questa tecnica considera il riconoscimento di volti come un problema di riconoscimento bidimensionale, sfruttando il fatto che le facce possono essere descritte da un piccolo insieme di viste bidimensionali. Le immagini del volto sono proiettate in uno spazio che ne codifica le variazioni. Tale spazio è definito dalle “autofacce”, ovvero gli autovettori dell'insieme di volti. Il riconoscimento avviene proiettando una nuova immagine nello spazio e confrontando la posizione della foto da verificare con quella dei volti conosciuti.

L'idea alla base di questo approccio è estrarre l'informazione rilevante per il riconoscimento dell'immagine di un volto considerando la variazione in un insieme di immagini del volto stesso. Si vuole trovare le principali componenti della distribuzione delle immagini di facce, ossia gli autovettori della matrice di covarianza dell'insieme delle immagini osservate. Gli autovettori descrivono la variazione tra le foto del volto e sono detti “autofacce”. Ogni immagine del volto è perciò una combinazione lineare delle “autofacce”. Si può pensare di approssimare i volti usando solo alcune “autofacce”: quelle, cioè, che hanno i maggiori autovalori, e che quindi descrivono la maggior variazione nell'insieme di immagini di un volto. In questo modo si migliora l'efficienza computazionale del metodo. Il riconoscimento avviene considerando la distanza euclidea tra l'autovettore calcolato dall'immagine da verificare e l'autovettore di una data faccia: si cerca il volto che minimizza tale distan-

za e se questa è al di sotto di una soglia prestabilita, la foto da verificare si ritiene corrispondente al volto in questione. La stessa tecnica può essere adottata per l'acquisizione di volti: si calcolano gli autovettori delle parti di un'immagine e si controlla se cadono nello spazio delle facce, ovvero lo spazio costituito dalle "autofacce". Si può esprimere ogni faccia come una combinazione lineare di queste "autofacce".

4.2 Rappresentazione delle caratteristiche locali

4.2.1 Preprocessamento

Prima di estrarre le caratteristiche locali da un insieme di immagini, si eseguono una serie di passi preliminari.

- Si utilizza il sistema 3D di individuazione delle orecchie proposto in [20]. Tale sistema fornisce in output un rettangolo di selezione ("Bounding Box" BB) dal quale una regione di interesse ("Region Of Interest" ROI) si estrae dall'immagine e può essere processata ai passi successivi.
- Per ridurre il rumore nelle immagini di input:
 - si applica un filtro mediano per eliminare i picchi,
 - si esegue un'interpolazione cubica per riempire eventuali mancanze di dati in determinati punti dell'immagine,
 - si applica un filtro gaussiano per "levigare" i dati.
- La superficie è normalizzata ad una posizione standard. Il baricentro della superficie è fatto corrispondere all'origine del sistema di riferimento. Sono calcolate le componenti principali (autovettori) corrispondenti ai due autovalori maggiori della superficie. La superficie è ruotata in modo tale che le due componenti principali siano allineate con gli assi x e y del sistema di riferimento.

4.3 Descrittore della caratteristica Histogram of Indexed Shapes (HIS)

Si possono caratterizzare gli oggetti in funzione dalle forma della loro superficie 3D. L'orecchio umano attorno all'elica o all'antielica contiene forme di

superfici a selle e a cresta, mentre la regione interna dell'orecchio è composta da solchi e canali.

4.3.1 Indice di forma e curvatura

Una misura quantitativa della forma della superficie in un punto \vec{p} chiamata indice di forma S_I è definita come da [21]:

$$S_I(\vec{p}) = \frac{1}{2} - \frac{1}{\pi} \arctan \left(\frac{k_{max}(\vec{p}) + k_{min}(\vec{p})}{k_{max}(\vec{p}) - k_{min}(\vec{p})} \right) \quad (4.1)$$

dove k_{min} e k_{max} sono le principali curvatures della superficie in \vec{p} con $k_{min} < k_{max}$ definite come:

$$k_{max}(\vec{p}) = H(\vec{p}) + \sqrt{H^2(\vec{p}) - K(\vec{p})} \quad (4.2)$$

$$k_{min}(\vec{p}) = H(\vec{p}) - \sqrt{H^2(\vec{p}) - K(\vec{p})} \quad (4.3)$$

dove $H(\vec{p})$ e $K(\vec{p})$ sono la curvatura media e la curvatura gaussiana rispettivamente.

Dalla definizione di S_I tutte le forme si possono mappare nell'intervallo $S_I = [0, 1]$. Ogni forma di superficie distinta corrisponde ad un unico valore di S_I eccetto che per le superfici piane; i vertici di superfici piane hanno un indeterminato indice di forma visto che $k_{min} = k_{max} = 0$. Il valore dell'indice di forma fornisce la nozione intuitiva della forma "locale" della superficie.

L'indice di forma di un oggetto rigido è indipendente dalla sua posizione, dal suo orientamento nello spazio ed anche dalla sua scala. Per codificare anche le differenze di scala si usa la curvatura, nota anche come energia di curvatura; da [21] si ha la formula:

$$C_v(\vec{p}) = \sqrt{\frac{k_{max}^2(\vec{p}) + k_{min}^2(\vec{p})}{2}} \quad (4.4)$$

che misura l'intensità della superficie di curvatura e descrive quanto lievemente o fortemente è curva la superficie.

4.3.2 Descrittore HIS

Per realizzare il descrittore ad istogramma occorre calcolare prima l'indice di curvatura e di forma per ogni punto della superficie che si andrà ad analizzare. Ogni punto fornirà un voto pesato, per un determinato intervallo dell'istogramma, basato sul valore dell'indice di forma con un peso che dipende dal suo indice di curvatura. I voti sono sommati all'interno degli opportuni intervalli uniformi dell'istogramma, formando il descrittore HIS. Il descrittore è poi normalizzato rispetto all'energia totale (o energia totale di curvatura).

4.4 Individuazione dei keypoint 3D

Per generale l'insieme di caratteristiche locali si cercano keypoint che siano robusti alle variazioni dell'immagine ed altamente distintivi. Il metodo di ricerca qui proposto è basato sul metodo proposto in [22] ma migliorato in quanto a robustezza e applicabilità su oggetti con curvature sporgenti come le orecchie. Nel metodo proposto in [22] la nuvola di punti dall'immagine è ottenuta campionando ad intervalli regolari. Ma osservando la struttura 3D delle immagini dell'orecchio la maggioranza dei punti significativi è situata in zone di superfici dell'orecchio contenenti elevati valori di curvatura. Quindi campionando tali zone otterremo con più probabilità keypoint ripetibili.

Invece di campionare uniformemente l'immagine candidata per ottenere i keypoint candidati, si propone di usare una finestra $b \times b$, con $b = 1mm$, per calcolare tali punti. Il punto centrale della finestra è indicato come keypoint candidato solo se il suo indice di curvatura è superiore a tutti gli indici dei punti contenuti nella finestra.

Quando si identifica un keypoint candidato si ritaglia dall'immagine un pezzo di superficie attorno al keypoint candidato; il ritaglio è delle dimensioni di una sfera centrata nel punto keypoint candidato. Se la sfera dovesse contenere keypoint candidati vicini al bordo dell'immagine questi sarebbero scartati. In tutti gli altri casi si usa la Principal Component Analysis (PCA)² alla superficie tagliata e sono calcolati gli autovalori e autovettori per computare il potenziale discriminativo.

Si accetta un keypoint candidato solo se gli autovalori calcolati sulla superficie tagliata soddisfano il criterio:

$$\frac{\lambda_3}{\sum_{i=1}^3 \lambda_i} > t_1 \quad \text{e} \quad \frac{\lambda_1}{\sum_{i=1}^3 \lambda_i} < t_2 \quad (4.5)$$

dove λ_1 e λ_3 sono il minore e maggiore autovalore rispettivamente. La soglia t_1 assicura che la regione tagliata associata al keypoint candidato abbia una certa somma di variazione di pendenza. La soglia t_2 assicura che il keypoint candidato non sia situato in una regione rumorosa o in un bordo dove le variazioni sono maggiormente in un'unica direzione. (Usualmente $t_1 = 0,01$ e $t_2 = 0,8$). Si mostra un esempio d'insieme nella Fig 4.2.

²PCA è un semplice metodo non-parametrico per estrarre informazione rilevante da un insieme di dati "confuso" (ridondante + rumoroso). Geometricamente l'obiettivo della PCA è presentare i dati nel riferimento che evidenzia maggiormente la loro struttura (Cambio di riferimento)

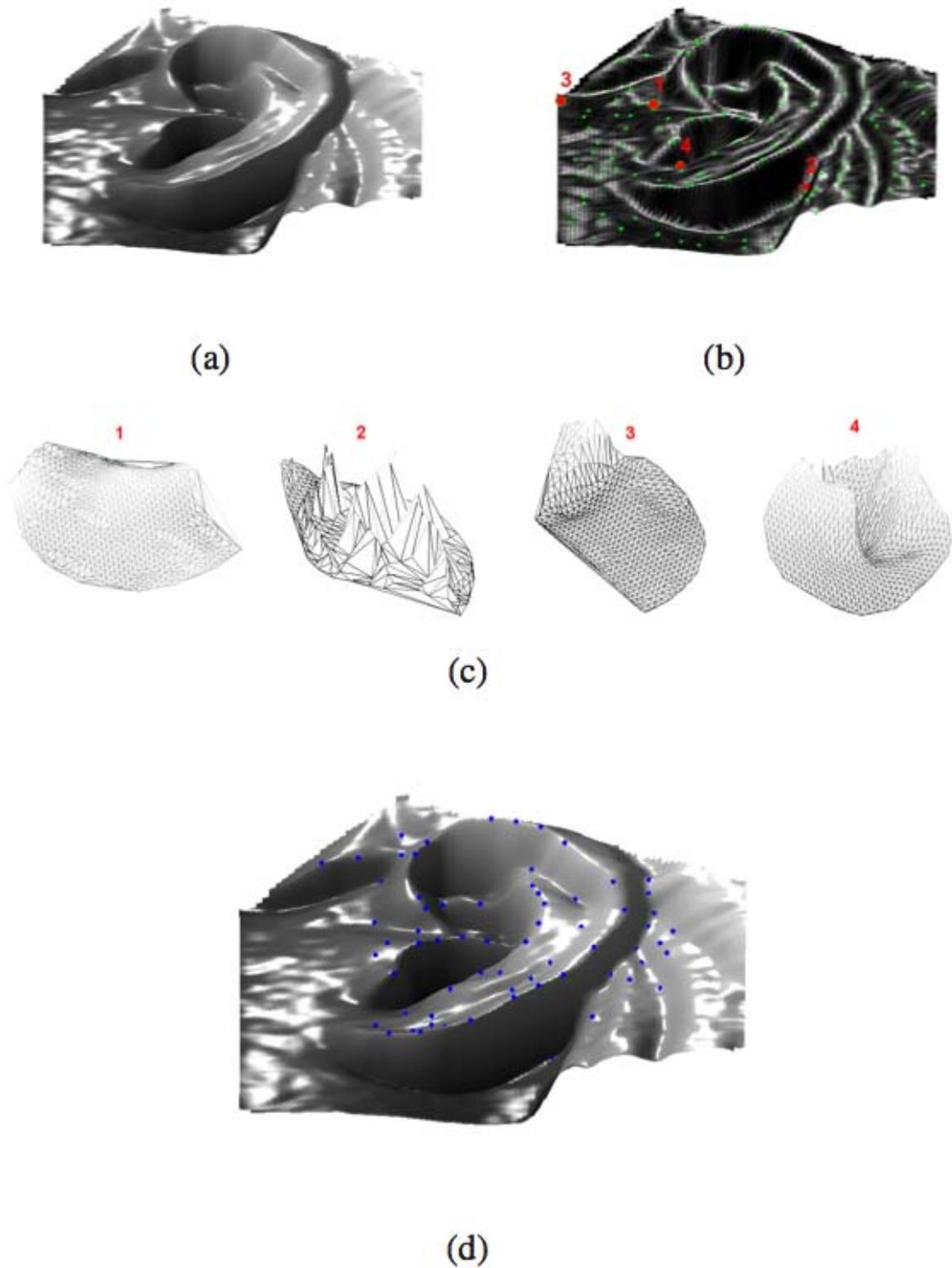


Figura 4.2: Individuazione dei keypoint: (a) Superficie iniziale. (b) Keypoint candidati. (c) PCA applicata alla superficie tagliata centrata sul keypoint. (d) Keypoint finali

Come prima cosa un insieme di keypoint candidati sono campionati sulla superficie basandosi sul loro indice di curvatura (Fig 4.2 (b)). Si applica la PCA ai punti nel vicinato dei keypoint candidati al fine di eliminare i keypoint candidati inadeguati e rumorosi. In Fig. 4.2 (c) si dimostra il passo della PCA in cui il keypoint candidato 1 (poco distintivo), il 2 (rumoroso) e il 3 (sul bordo) sono rigettati, mentre il 4 è mantenuto. Tutti i keypoint candidati mantenuti sono nella Fig. 4.2 (d).

Per dimostrare l'efficacia dell'algoritmo di estrazione dei keypoint, viene effettuato un esperimento ripetibile sui keypoint estratti da 200 immagini 3D dell'orecchio associate a 100 soggetti; ad ogni soggetto corrisponde una coppia di immagini dell'orecchio. Dal momento che le immagini sono ottenute dal mondo reale la "verità di terra"³ corrispondente ai keypoint è sconosciuta. Nell'esperimento si approssimerà "la verità di terra" con l'algoritmo di registrazione basato sull'ICP⁴ [22]. La coppia dei modelli d'orecchio dello stesso soggetto è registrata utilizzando tutti i punti ricavati da entrambi i modelli. Dato un keypoint e il suo vicinato nella prima immagine, il keypoint più vicino presente nella seconda immagine di controparte, dopo che è stata allineata (all'interno di uno stesso vicinato), è considerato come corrispondente. Quando la corrispondenza è all'interno di una certa distanza dal keypoint di controparte, allora quest'ultimo è detto ripetibile. La Fig. 4.3 mostra una percentuale cumulativa sulla ripetitività come funzione dell'incremento della distanza nella corrispondenza. La ripetitività raggiunge il 28,6% a 1mm campionando i punti con massimo indice di curvatura locale, mentre si ha il 20,7% ottenuto dal campionamento uniforme. Da notare che si considera la ripetitività ad una distanza minore della risoluzione delle immagini. Complessivamente il nostro algoritmo di acquisizione dei keypoint ottiene ripetitività migliori campionando con sugli indici di curvatura maggiori.

³Col termine "verità di terra" ci si riferisce alla correttezza della classificazione del training set nelle tecniche di apprendimento supervisionato. Un esempio di apprendimento supervisionato è il filtro bayesiano per lo spam: l'algoritmo prevede un apprendimento iniziale manuale per distinguere fra spam e non spam; imprecisioni sulla "verità di terra" si ripercuoteranno sul risultato finale dell'algoritmo.

⁴ICP è un algoritmo per minimizzare la differenza tra due nuvole di punti. ICP è spesso usato per ricostruire superfici 2D o 3D da diverse scansioni, per ottenere la pianificazione percorso ottimale ...

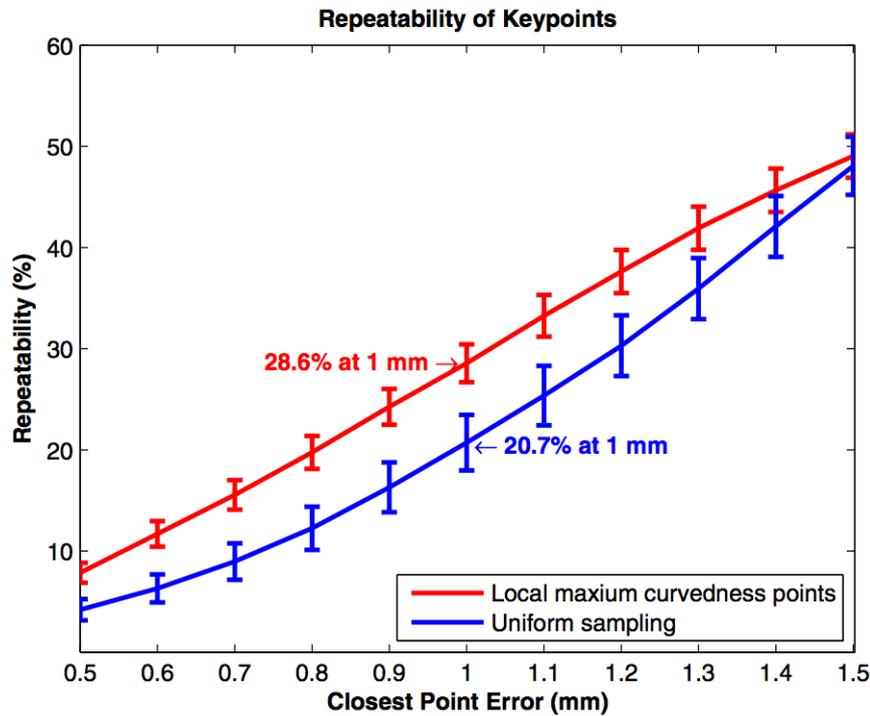


Figura 4.3: Ripetibilità dei keypoint nelle immagini dell'orecchio 3D.

4.5 Rappresentazione delle caratteristiche locali

Dalle posizioni dei keypoint individuati si può ottenere un sistema di coordinate ripetibile 3D che descrive la superfici locali dell'orecchio. Il passo successivo è costruire un descrittore della caratteristica che rappresenta la superficie locale dell'orecchio che è altamente distintiva, mentre rimane invariante agli altri cambiamenti come posizione, disordine di sfondo e rumore. La rappresentazione della caratteristica locale è un'estensione della caratteristica HIS introdotto nel 4.3.2. L'estensione contiene differenti metodi computativi che rendono la rappresentazione della caratteristica più accurata ed informativa, consentendo di individuare variazioni di forma più dettagliate all'interno dell'orecchio fra differenti soggetti.

4.5.1 Istogramma del taglio di superficie associata al descrittore di forma indicizzata (SPHIS)

Come detto nel 4.3.2, l'HIS può essere usato per codificare informazioni sulla forma di qualsiasi superficie. Si può formare un HIS di dimensione arbitraria uniformando i valori degli indici di forma in un range compreso fra $[0, 1]$. Tanto maggiore è la dimensione di un HIS, tanto meno esso è descrittivo ed aumenta la sensibilità al rumore. Basato sull'HIS, il descrittore SPHIS è usato per descrivere i keypoint ed è costruito dalla superficie tagliata attorno ai keypoint. Le Fig.4.4 e Fig.4.6 descrivono la procedura per ottenere l'SPHIS:

- si taglia la superficie dove è situato il keypoint nella misura di una sfera centrata sul keypoint avente raggio r .

Il raggio r determina la località del taglio di superficie e consente di preferire distinguibilità o robustezza. Minore è il valore di r meno distinguibile è il taglio di superficie ma è più resistente alle variazioni di posizione e al disordine di sfondo. La scelta di r dipende dall'oggetto a cui si applica il taglio; nel nostro metodo per il riconoscimento 3D dell'orecchio $r = 14mm$ ed è determinato in base alla dimensione dell'orecchio.

- I punti contenuti all'interno del taglio di superficie sono ulteriormente divisi in 4 sottoinsiemi utilizzando altre 3 sottosfere concentriche con raggio $r_i = \frac{i \cdot r}{4}$ con $i = 1, 2, 3$ centrate sempre nel keypoint formando 4 tagli di sotto-superfici come mostrato nella prima riga di Fig 4.6. Si è divisa ulteriormente la superficie tagliata dalla sfera al fine di ottenere maggiori informazioni spaziali su essa.

Dopo aver diviso le quattro sotto-superfici adiacenti, si costruisce un descrittore HIS per ognuna di esse; si utilizzano gli indici di curvatura di ogni punto delle sotto-superfici e si inseriscono all'interno degli intervalli dell'indice di forma come indicato nel paragrafo 4.3.2. Il descrittore SPHIS forma un array di 4 descrittori HIS con 16 intervalli (16 forme indicizzate), uno per sotto-superficie, dove l'altezza di ogni intervallo corrisponde all'importanza di ogni intervallo nell'istogramma (Fig. 4.6). I 4 descrittori HIS sono concatenati a formare un vettore di 64 celle. Infine il valore dell'indice di forma del keypoint è concatenato al vettore al fine di aumentare la sua discriminabilità ed evitare che keypoint diversi possano essere "matchati" erroneamente nei passi successivi. Quindi si ottiene un array di 65 elementi per rappresentare il taglio della superficie locale.

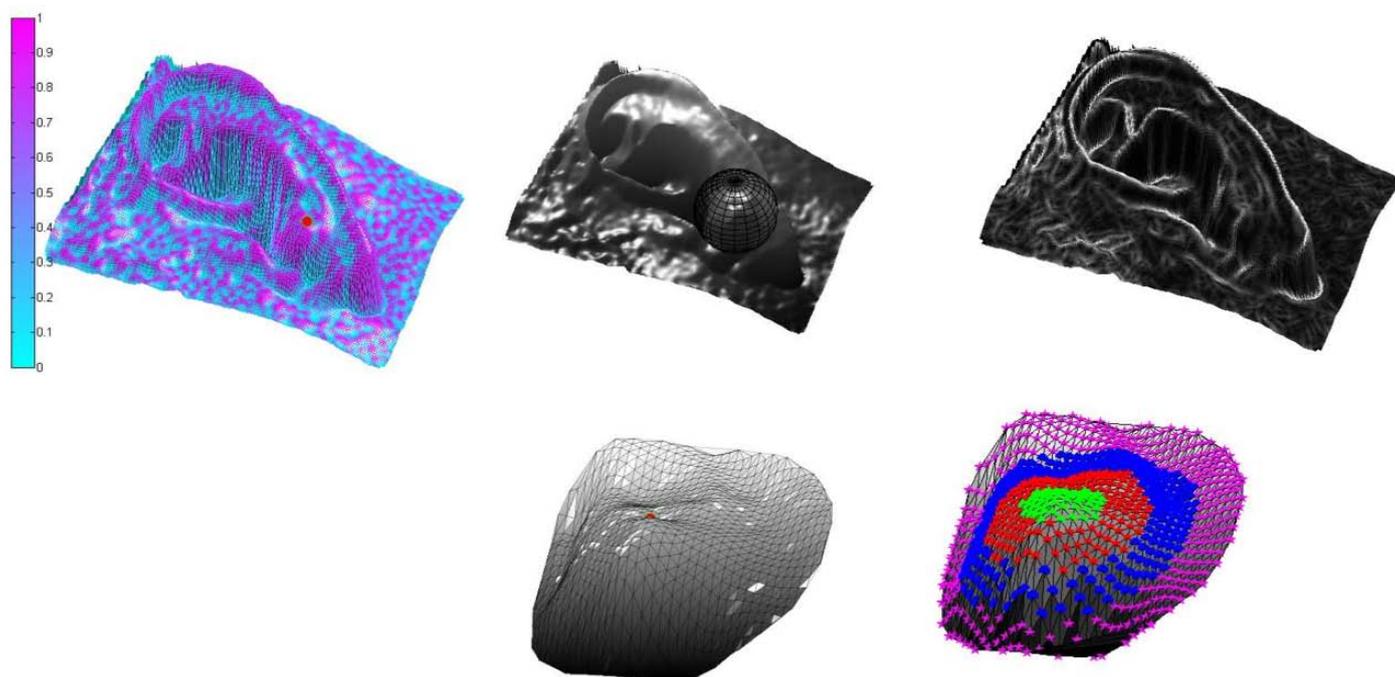


Figura 4.4: Estrazione dell' SPHIS. Prima riga da sinistra a destra: la mappa degli indici di forma, l'orecchio 3D con la sfera centrata sul keypoint usato per tagliare la superficie al fine di realizzare il SPHIS, la mappa di curvatura. Seconda riga da sinistra a destra: un taglio di superficie dalla sfera associata al keypoint, quattro diverse ampiezza associate allo stesso taglio precedente.

4.6 Matching delle superfici locali

Nella rappresentazione della caratteristica locale, la superficie 3D dell'orecchio è descritta da un insieme sparso di keypoint, e, associato ad ogni keypoint, c'è un descrittore SPHIS della caratteristica che codifica le informazioni della superficie locale in un sistema di coordinate oggetto-centrico. L'obiettivo del motore di matching delle caratteristiche locali è fare combaciare questi keypoint al fine di far combaciare l'intera superficie.

Per consentire un efficiente matching fra galleria e modelli d'indagine, tutte le immagini della galleria sono processate per prime. I keypoint estratti e i loro rispettivi descrittori di caratteristica SPHIS sono memorizzati nella galleria. Ogni caratteristica rappresenta un'informazione sulla superficie locale in modo che sia invariante alle trasformazioni della superficie stessa. Un tipico orecchio 3D produce circa 100 caratteristiche sovrapposte che formano una rappresentazione ridondante della superficie di partenza.

Al passo di matching delle caratteristiche locali, data un'immagine d'in-

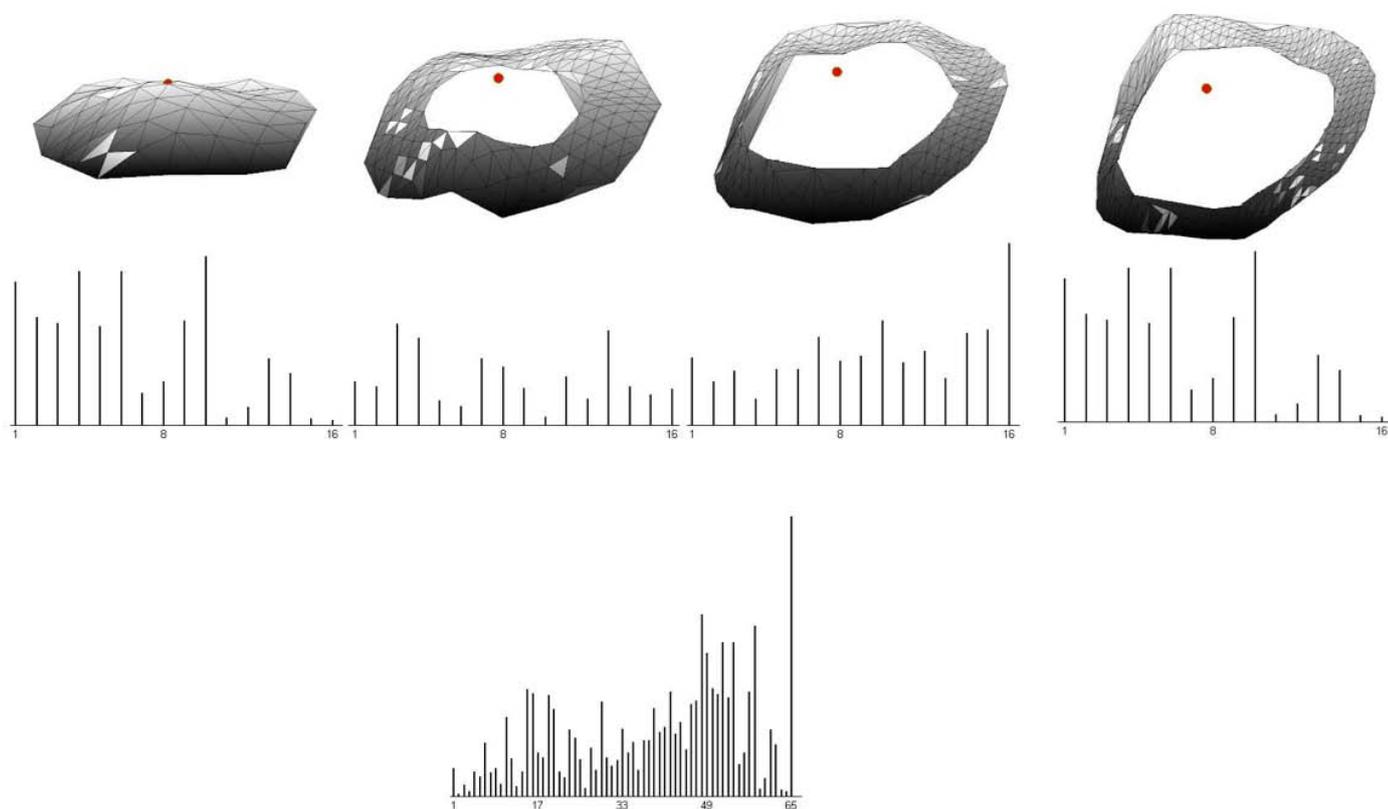


Figura 4.5: Estrazione dell' SPHIS. Prima riga da sinistra a destra: la mappa degli indici di forma, l'orecchio 3D con la sfera centrata sul keypoint usato per tagliare la superficie al fine di realizzare il SPHIS, la mappa di curvatura. Seconda riga da sinistra a destra: un taglio di superficie dalla sfera associata al keypoint, quattro diverse ampiezza associate allo stesso taglio precedente.

indagine, si estrae un insieme di keypoint coi rispettivi SPHIS descrittori utilizzando gli stessi parametri usati nell'estrazione delle caratteristiche nella galleria di immagini. Per ogni caratteristica nell'immagine d'indagine, si determina la caratteristica più vicina ad essa in riferimento all'immagine della galleria; viene calcolata la distanza euclidea fra i descrittori della caratteristica. Si applica una soglia $t = 0, 1$ per scartare le caratteristiche non hanno un match adeguato. Tale procedura è ripetuta per ogni keypoint d'indagine risultante dall'insieme di keypoint corrispondenti. Le corrispondenze non corrette sono filtrate usando vincoli geometrici. Introduciamo il metodo iterativo ortogonale Procuste⁵ (descritto nel listato seguente) per allineare i due

⁵L'analisi di Procuste consiste nel sovrapporre due o più forme e, tramite rotazio-

insiemi di keypoint ed eliminare le corrispondenze errate valutando la consistenza geometrica. Dopo aver applicato tale metodo, il motore di matching delle superfici locali restituisce il numero dei keypoint corrispondenti M come risultato per ogni coppia immagine di galleria e immagine d'indagine simile. Fig.4.6 mostra un esempio di corrispondenze di keypoint trovate da una coppia di modello d'orecchio dalla galleria e modello d'orecchio d'indagine.

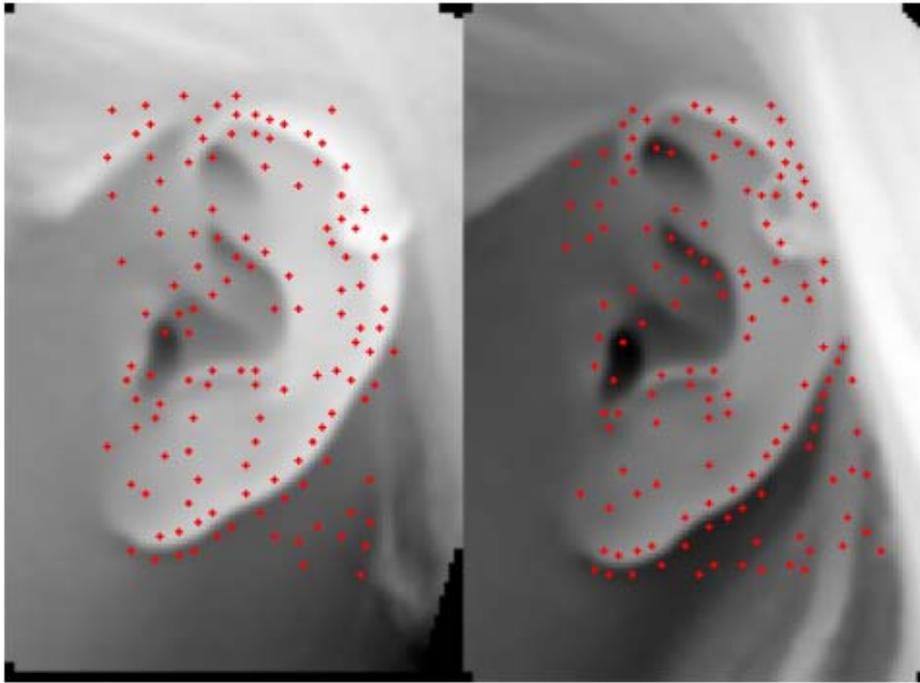
```

1) Vi sia un insieme di  $M$  keypoint iniziali corrispondenti.
   Siano i punti della galleria  $\vec{g}_i = (x_i^g, y_i^g, z_i^g)^T$  e
   siano i punti d'indagine  $\vec{p}_i = (x_i^p, y_i^p, z_i^p)^T$  con  $i = 1, \dots, M$ 
2) repeat
3) Allineo i keypoint del modello di galleria e d'indagine
   • Calcolo il baricentro dei keypoint dell'immagine di
   indagine e della galleria:  $\vec{g}_c = \frac{1}{M} \sum_i^M \vec{g}_i$ ,  $\vec{p}_c = \frac{1}{M} \sum_i^M \vec{p}_i$ 
   • Calcolo la matrice di rotazione  $\mathbf{R}$  usando la decomposizione
   ai valori singolari:  $\mathbf{C} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$ ,  $\mathbf{R} = \mathbf{V}\mathbf{U}^T$  con
    $\mathbf{C} = \frac{1}{M} \sum_i^M (\vec{p}_i - \vec{p}_c)(\vec{g}_i - \vec{g}_c)^T$ 
   • Ottengo il vettore di transizione  $\vec{t} = \vec{g}_c - \mathbf{R}\vec{p}_c$ 
   • Allineo i keypoint del modello d'indagine e della galleria
   usando  $\mathbf{R}, \vec{t}$ :  $\vec{p}'_i = \mathbf{R}\vec{p}_i + \vec{t}$ 
   • Aggiorno le distanze dei keypoint:  $d_i = \|\vec{g}_i - \vec{p}'_i\|_2$ 
   • Trovo il valore maggiore di  $d_i$ . Se  $d_{max} > 1.5mm$  la corri-
   spondenza è rimossa e  $M \leftarrow M - 1$ 
4) until  $d_{max} < 1.5mm$  o  $M < 3$ 
5) output di  $M$  come risultato di similarità

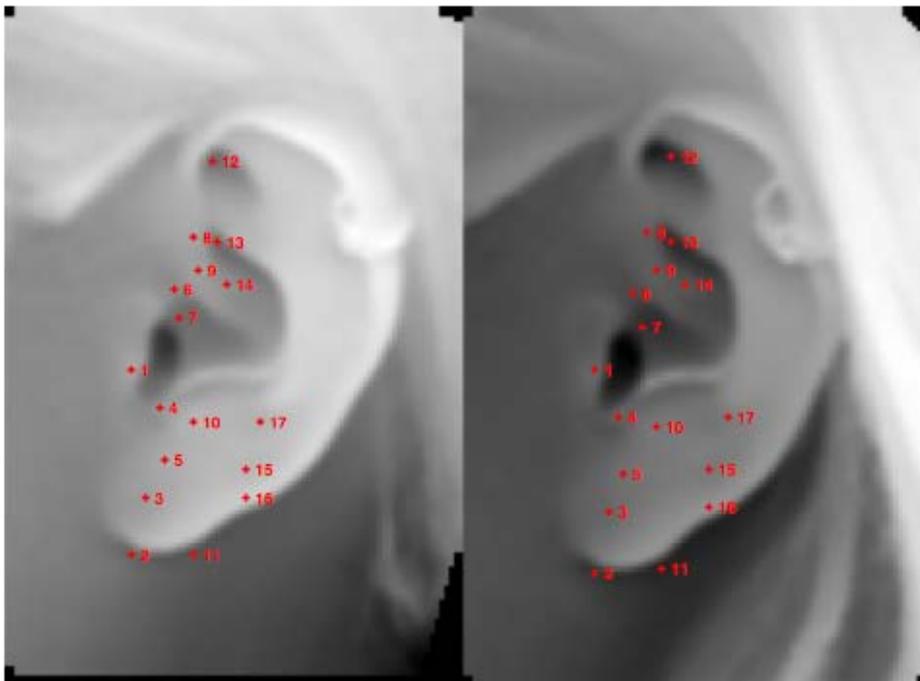
```

Listing 4.1: Analisi di Procruste ortogonale e iterativa per l'eliminazione delle non corrispondenze

ni, traslazioni e modificandone le dimensioni, arrivare ad ottenere una sovrapposizione ottimale.



(a)



(b)

Figura 4.6: Esempio di ricerca delle corrispondenze delle caratteristiche per una coppia immagine di galleria e immagine di indagine dello stesso soggetto. (a) Keypoint trovati sulle orecchie. (b) Corrispondenze delle caratteristiche trovate col motore di matching delle superfici locali.

4.7 Estrazione della caratteristica olistica

4.7.1 Preprocessamento

La precedente sezione descrive un metodo per stabilire una connessione fra la coppia di immagine galleria e d'indagine. Il modello d'indagine è registrato sul modello della galleria applicando la trasformazione ottenuta al passo del matching locale. Nel caso in cui il numero di corrispondenze trovato sia minore di tre, si ritorna al passo sulla normalizzazione della posizione descritto alla sezione 4.2.1 in cui si descriveva la registrazione del modello e si modificano i parametri di registrazione del modello.

4.7.2 Voxelizzazione della superficie

La rappresentazione olistica utilizzata è la voxelizzazione della superficie. Il motivo dell'utilizzo dell'operatore di Voxel è per utilizzare metodi alternativi e più efficienti piuttosto che usare l'errore quadratico medio tra il modello d'indagine registrato e il modello della galleria. L'uso dell'errore quadratico medio è molto in voga nella letteratura, ma è una tecnica computazionalmente dispendiosa in quanto richiede i punti più vicini della superficie nella sua immagine di controparte (la complessità di ricerca dei punti più vicini è $O(N_g N_p)$ dove N_g e N_p denotano il numero di punti compresi nel modello della galleria e nel modello d'indagine rispettivamente).

La voxelizzazione è un processo per approssimare una superficie continua in un dominio discreto 3D [23]. Il processo utilizza un array di elementi di volume (i voxel) in uno spazio 3D. Un voxel è analogo ad un pixel che rappresenta un'immagine a 2D. Il vantaggio di tale rappresentazione è la robustezza al rumore di superficie che può accadere quando c'è della specularità sulla superficie durante l'acquisizione. La sua robustezza al rumore è consentita dalla possibilità di variare il passo di quantizzazione, (ossia la dimensione del voxel), usato per rendere discreta la superficie. Inoltre la voxelizzazione fornisce una rappresentazione concisa della superficie, (a seconda della dimensione del voxel usata), riducendo la dimensione necessaria per la memorizzazione nel database. Ancora la voxelizzazione produce rappresentazioni normalizzate e a dimensione fissa anche per insiemi di oggetti diversi; ciò consente una efficiente comparazione fra rappresentazioni voxel (per esempio calcolando il prodotto scalare fra rappresentazioni). Inoltre la voxelizzazione può codificare attributi come la presenza (se un punto nella superficie è contenuto nel voxel), la densità (il numero di punti contenuti nel voxel) e caratteristiche di superficie (la curvatura media dei punti contenuti nel voxel).

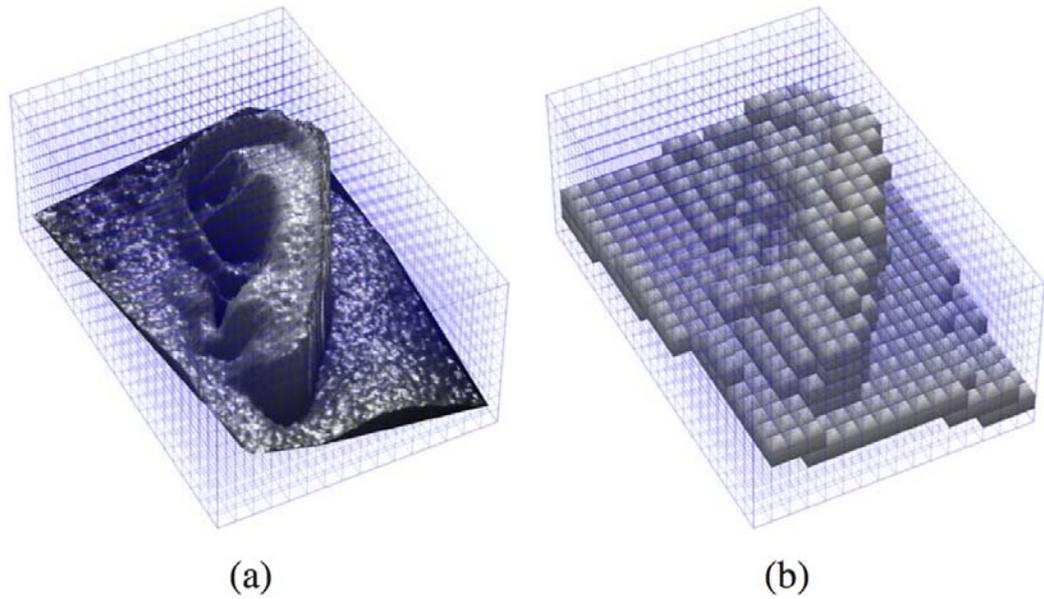


Figura 4.7: Voxelizzazione binaria. a) Modello d'orecchio campionato incluso in una griglia di voxel cubici di $4.0mm$. b) Il modello voxelizzato.

La voxelizzazione usata nel metodo è quella binaria come mostrato nella Fig. 4.7. Questa rappresentazione semplicemente codifica la presenza di un punto all'interno di un voxel. Un voxel che contiene al suo interno un punto ha valore "1" oppure "0" altrimenti. Successivamente si descrive l'algoritmo il processo di voxelizzazione usando gli assunti precedenti. L'input dell'algoritmo son i punti della superficie da voxelizzare $\{\vec{\mathbf{p}}_i\}_{i=1}^N$, le dimensioni del voxel $\{r_x, r_y, r_z\}$ e l'estensione spaziale della griglia $\{x_{lo}, y_{lo}, z_{lo}, x_{hi}, y_{hi}, z_{hi}\}$ di voxel. La variabile ϵ è usata per assicurarsi che i punti sul bordo della griglia di voxel siano assegnati ai voxel; il valore di ϵ deve essere maggiore di 0 e minore della dimensione minima del voxel ($\epsilon = 1 \cdot 10^{-15}$). Segue l'algoritmo di voxelizzazione binaria:

- 1) Dati i punti della superficie $\{\vec{\mathbf{p}}_i\}_{i=1}^N = \{x_i, y_i, z_i\}_{i=1}^N$ con $\{r_x, r_y, r_z\}$ dimensioni del voxel e $\{x_{lo}, y_{lo}, z_{lo}, x_{hi}, y_{hi}, z_{hi}\}$ estensione spaziale:
- 2) Inizializzo $\mathbf{V} = [v_{i,j,k}]_{s_x \times s_y \times s_z} = \mathbf{0}$ dove $\forall d \in \{x, y, z\}$:
 $s_d = \lceil (d_{hi} + \epsilon - d_{lo}) / r_d \rceil$
- 3) for $i = 1, \dots, N$ do
- 4) $v^{\psi_x(x_i), \psi_y(y_i), \psi_z(z_i)} = 1$, dove:
 $\psi_d(d_i) = \lfloor (d_i - d_{lo}) / r_d \rfloor + 1$
- 5) end for

Listing 4.2: Voxelizzazione binaria

4.7.3 Motore di matching per la superficie olistica

Al passo di registrazione (offline) della galleria, per un dato modello della galleria, a partire dal rettangolo di selezione contenente il modello è costruita una griglia di voxel. Il modello nella griglia è voxelizzato e la sua rappresentazione è registrata nella galleria. Al passo online di matching locale la trasformazione usata per registrare la coppia modello di galleria modello d'indagine, è applicata al rettangolo di selezione del modello d'indagine. Occorre, quindi, estendere la griglia di voxel anche al modello d'indagine ampliandola delle dimensioni del rettangolo di selezione del modello d'indagine. La griglia di voxel estesa è usata per voxelizzare il modello d'indagine. Inoltre la rappresentazione voxelizzata del modello della galleria tiene conto dell'estensione riempiendo di zeri i punti mancanti. Si tenga conto che entrambi i modelli sono voxelizzati usando la medesima griglia di voxel. Voxelizzando entrambi i modelli con la stessa griglia di voxel e vettorizzando le voxelizzazioni, si producono vettori con le stesse dimensioni. Le similitudini fra questi vettori è calcolata usando la misura del coseno di similitudine⁶ data da:

$$S(p, g) = \frac{\vec{\mathbf{v}}_p \vec{\mathbf{v}}_g}{\|\vec{\mathbf{v}}_p\| \|\vec{\mathbf{v}}_g\|} \quad (4.6)$$

dove $\vec{\mathbf{v}}_p$ e $\vec{\mathbf{v}}_g$ denotano la versione vettorizzata della matrice \mathbf{V} (presentata nell'algoritmo 4.2) del modello d'indagine e del modello della galleria rispettivamente. Sebbene certi valori dei voxel sono 0 essi non si inficiano il calcolo della 4.6.

Si è stabilita la dimensione ottimale del voxel per via empirica attraverso successivi esperimenti. Si sono considerati solo voxel cubici. Dagli esperimenti si è dimostrato che la dimensione di $1mm$ dei voxel produce le migliore performance di riconoscimento.

4.7.4 Fusione

I componenti del match delle caratteristiche locali ed olistiche sono contenuti nelle matrici indipendenti di similarità S_i ognuna di dimensione $P \times G$ dove $i \in \{1, 2\}$ denota il motore di matching e P e G rappresenta il numero di modelli d'indagine e di galleria rispettivamente. Si fonderanno i risultati locali ed olistici attraverso la tecnica della somma pesata. Comunque la combinazione dei risultati di matching è eloquente solo quando i risultati degli

⁶Il coseno di similitudine, o cosine similarity, è una tecnica euristica per la misurazione della similitudine tra due vettori effettuata calcolando il coseno tra di loro.

accoppiamenti individuali sono comparabili. Perciò la funzione sigmoidea⁷, usata per la normalizzazione del risultato, è una tecnica efficiente e robusta usata per trasformare i risultati ottenuti da diversi accoppiamenti in un dominio comune. È definita come:

$$s_j^n = \begin{cases} \frac{1}{1+\exp\left(-2\left(\frac{s_j-\tau}{\alpha_1}\right)\right)} & \text{se } s_j < \tau, \\ \frac{1}{1+\exp\left(-2\left(\frac{s_j-\tau}{\alpha_2}\right)\right)} & \text{altrimenti.} \end{cases} \quad (4.7)$$

dove s_j e s_j^n sono i risultati prima e dopo la normalizzazione, τ è il punto operativo di riferimento e α_1 e α_2 sono il limite sinistro e destro della regione in cui la funzione è lineare. Lo schema a doppia normalizzazione della sigmoide trasforma il risultato in un valore nell'intervallo compreso fra $[0, 1]$, e i risultati al di fuori dei due limiti (α_1 e α_2) sono trasformati non linearmente in modo da ridurre l'influenza di tali risultati alle estremità della distribuzione. Nella implementazione del metodo si è scelto τ , α_1 e α_2 in modo che τ , $\tau - \alpha_1$ e $\tau + \alpha_2$ corrispondano ai valori associati ai risultati di match genuino fino al 60% 95% e 5% rispettivamente. La somma pesata dei risultati normalizzati è usata per generare il risultato finale:

$$S_f = \sum_{j=1}^2 w_j s_j^n \quad (4.8)$$

dove s_j^n e w_j sono il risultato di match normalizzato e il peso della j -esima modalità rispettivamente, con la condizione $\sum_{j=1}^2 w_j = 1$. I pesi possono essere assegnati per ogni accoppiamento da una ricerca basata sulle performance individuali. In questo metodo scegliamo semplicemente $w_j = 0.5$ $j = \{1, 2\}$ sia per i gli accoppiamenti locali che olistici.

4.8 Risultati

4.8.1 Dati

Per valutare le prestazioni del sistema si è utilizzato la collezione G del dataset biometrico sull'orecchio 3D pubblico dell'università di Notre Dame (UND). Il dataset comprende 1801 immagini di profili appartenenti a 415 soggetti.

⁷La funzione sigmoidea è una funzione matematica che produce una curva sigmoide; una curva avente un andamento ad "S". Spesso, la funzione sigmoide si riferisce ad uno speciale caso di funzione logistica e definita dalla formula: $P(t) = \frac{1}{1+e^{-t}}$

4.8.2 Prestazioni di riconoscimento

Per l'identificazione l'approccio di matching proposto, di combinazione delle caratteristiche delle superfici olistiche e locali, ottiene un rapporto di riconoscimento rank-1⁸ di 98.6% su un insieme di 415 soggetti con 415 modelli d'indagine e 415 modelli di galleria. Inoltre ottiene un rapporto di ugual errore (EER) di 1.6%. Nella Fig. 4.8 sono rappresentate le curve CMC per le modalità e la fusione delle caratteristiche. Nella Fig. 4.9 sono rappresentate le curve ROC per le modalità e la fusione delle caratteristiche.

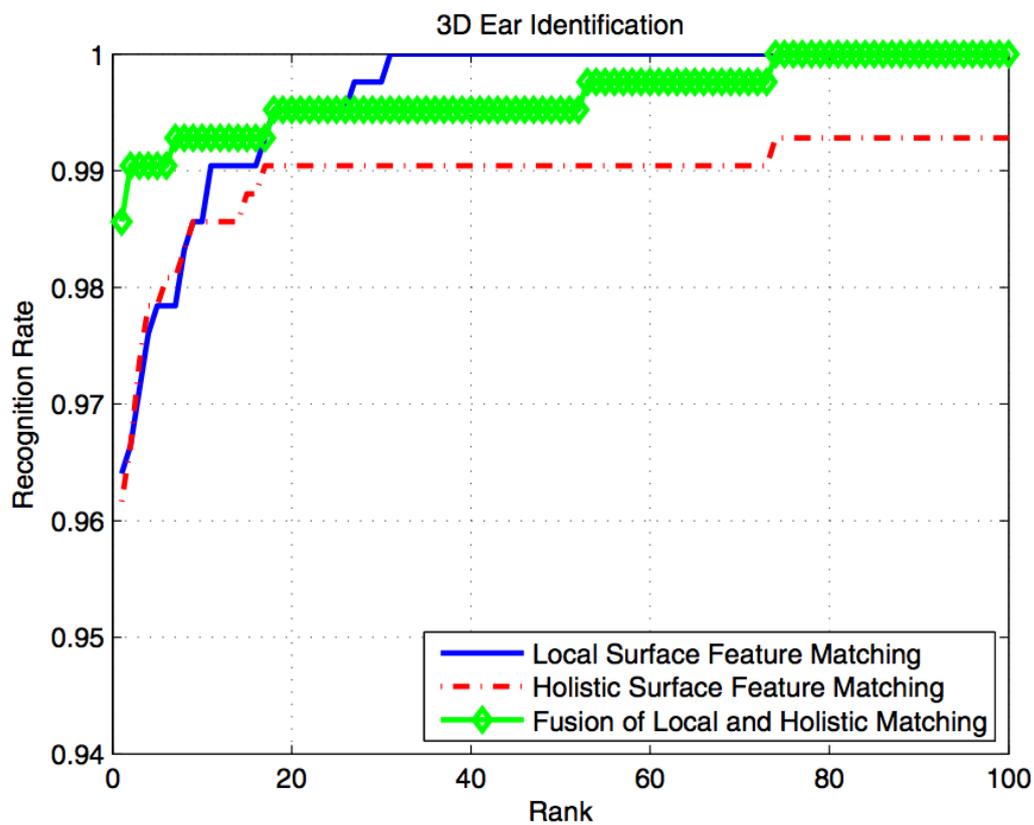


Figura 4.8: Comparazione delle performance del sistema di riconoscimento delle immagini 3D dell'orecchio esterno come curve CMC

⁸Per un immagine d'indagine l'algoritmo restituisce n risultati di similarità che corrispondono a n immagini nella galleria. Per un riconoscimento di rank- k l'algoritmo restituisce le etichette associate alle immagini di galleria corrispondenti ai k valori maggiori ottenuti dal risultato di similarità.

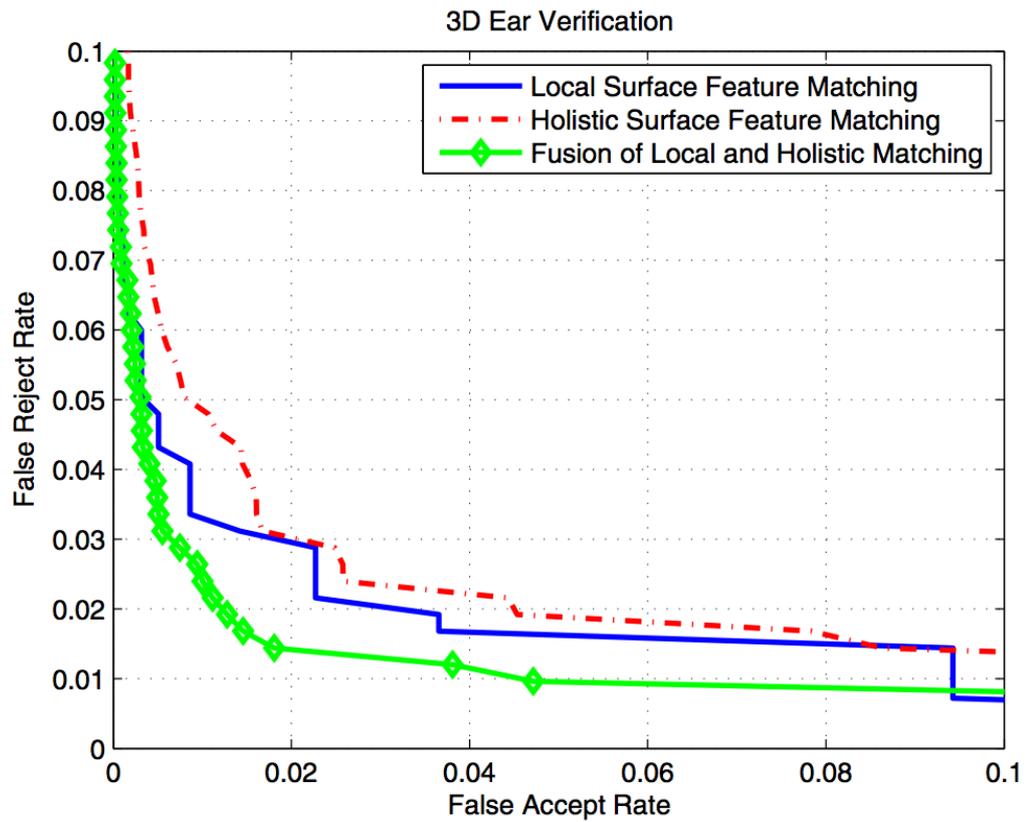


Figura 4.9: Comparazione delle performance del sistema di riconoscimento delle immagine 3D dell'orecchio esterno come curve ROC

Elenco delle figure

1	Orecchio esterno	1
1.1	Riflessione e rifrazione della luce fra due mezzi diversi m_1 e m_2 [15]	7
1.2	Riflessione totale della luce [15]	9
1.3	Convoluzione bidimensionale	10
1.4	Schema dell'operazione di convoluzione	12
1.5	Funzione di trasferimento Gaussiana h	12
1.6	Applicazione di h all'immagine f	13
1.7	Nuova immagine g ottenuta dalla convoluzione	13
1.8	Profilo di una discontinuità in termini di gradiente e sua rappresentazione grafica	14
1.9	il vettore normale usato per il calcolo delle rifrazione o della riflessione	21
1.10	esempio del corso di una raggio in un immagine 4×4	23
1.11	Passi per l'individuazione dell'orecchio	25
2.1	Divisione dell'intorno di un pixel in 4 regioni evidenziate dai colori giallo, verde, rosso e blu.	30
2.2	Pixel marcati nelle 2 immagini sogliate N_H e N_L e uniti se connessi	31
2.3	Mappa dei bordi	35
2.4	Grafo di connettività dei bordi	36
2.5	Orecchio individuato	37
2.6	Funzione di distribuzione di 200 campioni d'orecchio e 200 campioni di non-orecchio	37
3.1	Maschere 3×3 per l'estrazione dei gradienti direzionali	43
3.2	Laplaciano di una funzione monodimensionale	44
3.3	Esempio di miglioramento dell'immagine. (a) Immagine originale. (b)ADHist. (c) NLM. (d) SF.	48
3.4	Diagramma a blocchi dell'algoritmo proposto	50

3.5	Esempi del database IIT Kanpur	53
3.6	Esempi del database dell'università di Notre Dame Collection E	54
3.7	Curve ROC per i data set del database IITK con varie combinazioni di tecniche di miglioramento dell'immagine	57
3.8	Curve ROC per UND-E con varie combinazioni di tecniche di miglioramento dell'immagine	58
4.1	Visione d'insieme del metodo	62
4.2	Individuazione dei keypoint: (a) Superficie iniziale. (b) Keypoint candidati. (c) PCA applicata alla superficie tagliata centrata sul keypoint. (d) Keypoint finali	67
4.3	Ripetitività dei keypoint nelle immagini dell'orecchio 3D.	69
4.4	Estrazione dell' SPHIS. Prima riga da sinistra a destra: la mappa degli indici di forma, l'orecchio 3D con la sfera centrata sul keypoint usato per tagliare la superficie al fine di realizzare il SPHIS, la mappa di curvatura. Seconda riga da sinistra a destra: un taglio di superficie dalla sfera associata al keypoint, quattro diverse ampiezza associate allo stesso taglio precedente.	71
4.5	Estrazione dell' SPHIS. Prima riga da sinistra a destra: la mappa degli indici di forma, l'orecchio 3D con la sfera centrata sul keypoint usato per tagliare la superficie al fine di realizzare il SPHIS, la mappa di curvatura. Seconda riga da sinistra a destra: un taglio di superficie dalla sfera associata al keypoint, quattro diverse ampiezza associate allo stesso taglio precedente.	72
4.6	Esempio di ricerca delle corrispondenze delle caratteristiche per una coppia immagine di galleria e immagine di indagine dello stesso soggetto. (a) Keypoint trovati sulle orecchie. (b) Corrispondenze delle caratteristiche trovate col motore di matching delle superfici locali.	74
4.7	Voxelizzazione binaria. a) Modello d'orecchio campionato incluso in una griglia di voxel cubici di $4.0mm$. b) Il modello voxelizzato.	76
4.8	Comparazione delle performance del sistema di riconoscimento delle immagine 3D dell'orecchio esterno come curve CMC	79
4.9	Comparazione delle performance del sistema di riconoscimento delle immagine 3D dell'orecchio esterno come curve ROC	80

Elenco delle tabelle

1.1	Indici di rifrazione misurati con una luce di lunghezza d'onda nel vuoto pari a 589 nm. [15]	6
2.1	Accuratezza nella registrazione dell'orecchio 3D sul database UND-J2	39
3.1	Prestazioni ottenute sul database IITK data set 1 per varie combinazioni di tecniche di miglioramento. In grassetto i risultati migliori.	58
3.2	Prestazioni ottenute sul database IITK data set 2 per varie combinazioni di tecniche di miglioramento. In grassetto i risultati migliori.	59
3.3	Prestazioni ottenute sul database UND-E per varie combinazioni di tecniche di miglioramento. In grassetto i risultati migliori.	59
3.4	Prestazioni comparate con le tecniche note [25] e [26] svolte sul database UND-E. In grassetto i risultati migliori.	59

Bibliografia

- [1] Ear biometrics: a survey of detection, feature extraction and recognition methods, A. Pflug C. Busch, Hochschule Darmstadt – CASED, Haardtring 100, 64295 Darmstadt, Germany, 2012
- [2] Abaza, A., Ross, A.: ‘Towards understanding the symmetry of human ears: a biometric perspective’. Fourth IEEE Int. Conf. on Biometrics: Theory Applications and Systems (BTAS), 2010
- [3] Bertillon, A.: ‘La Photographie Judiciaire: Avec Un Appendice Sur La Classification Et L’Identification Anthropometriques’ (Gauthier-Villars, Paris, 1890)
- [4] Imhofer, R.: ‘Die Bedeutung der Ohrmuschel für die Feststellung der Identität’, Archiv für die Kriminologie, 1906, 26, pp. 150–163
- [5] Iannarelli, A.V.: ‘Ear identification’ (Paramont Publishing Company, 1989)
- [6] Sforza, C., Grandi, G., Binelli, M., Tommasi, D.G., Rosati, R., Ferrario, V.F.: ‘Age- and sex-related changes in the normal human ear’, Forensic Sci. Int., 2009, 187, (1 – 3), pp. 110e1 – 110e7 Available from: <http://www.sciencedirect.com/science/article/pii/S0379073809000966>
- [7] Meijerman, L., Van Der Lugt, C., Maat, G.J.R.: ‘Cross-sectional anthropometric study of the external ear’, J. Forensic Sci., 2007, 52, (2), pp. 286 – 293
- [8] Ibrahim, M.I.S., Nixon, M.S., Mahmoodi, S.: ‘The effect of time on ear biometrics’. Int. Joint Conf. on Biometrics (IJCB), 2011, pp. 1–6
- [9] Alberink, I., Ruifrok, A.: ‘Performance of the FearID earprint identification system’, Forensic Sci. Int., 2007, 166, (2 – 3), pp. 145 – 154

- [10] Alistair, H., Cummings, A.H., Nixon, M.S., Carter, J.N.: ‘A novel ray analogy for enrolment of ear biometrics’. Fourth IEEE Int. Conf. on Biometrics: Theory Applications and Systems (BTAS), 2010
- [11] A. H. Cummings, M. S. Nixon, and J. N. Carter, “Circle detection using the image ray transform,” in Int’l Conf. Computer Vision Theory and Applications (VISAPP 2010), 2010.
- [12] Alastair H. Cummings, Mark S. Nixon, John N. Carter, ‘The Image Ray Transform for Structural Feature Detection’, School of Electronics and Computer Science, University of Southampton, Southampton, SO17 1BJ, United Kingdom, 2011
- [13] Pnueli, Y., Bruckstein, A., 1994. Digidürer - a digital engraving system. *The Visual Computer* 10, 277–292.
- [14] Meyer, F., Maragos, P., 1999. Multiscale morphological segmentations based on water-shed, flooding, and eikonal pde. *Lecture Notes in Computer Science* 1682, 351–362.
- [15] J.D. Cutnell, K.W. Johnson, *Elementi di fisica*, Zanichelli editore S.p.A., 2010
- [16] Hill, F., 2000. *Computer graphics using OpenGL*. Prentice Hall. chapter 12. 3rd edition. p. 678.
- [17] <http://www.ino.it/home/cosimo/ttr/>
- [18] UND Profile Face Database, Collection J2. <http://www.nd.edu/cvrl/CVRL/DataSets.html>.
- [19] K. Messer, J. Matas, J. Kittler, J. Luetin, and G. Maitre, “XM2VTSDB: The extended M2VTS database,” in Proc. 2nd Int’l Conf. Audio and Video-based Biometric Person Authentication, 1999, pp. 965–966.
- [20] J. Zhou, S. Cadavid, and M. Abdel-Mottaleb. Histograms of categorized shapes for 3d ear detection. In *IEEE Fourth International Conference on Biometrics: Theory, Applications and Systems*, September 2010.
- [21] C. Dorai and A. Jain. Cosmos-a representation scheme for 3d free-form objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(10):1115–1130, 1997

- [22] A. Mian, M. Bennamoun, and R. Owens. Keypoint detection and local feature matching for textured 3d face recognition. *International Journal of Computer Vision*, 79(1):1–12, 2008.
- [23] S.W.Wang and A.E.Kaufman. Volume sampled voxelization of geometric primitives. In *Proceedings of the 4th conference on Visualization*, pages 78–84, 1993.
- [24] Prakash, S., Jayaraman, U., Gupta, P. (2009). Connected component based technique for automatic ear detection. In *Proc. of 16th IEEE int'l conference on image processing (ICIP' 09)*, Cairo, Egypt (pp. 2741–2744).
- [25] Nanni, L., Lumini, A. (2007). A multi-matcher for ear authentication. *Pattern Recognition Letters*, 28(16), 2219–2226.
- [26] Nanni, L., Lumini, A. (2009). Fusion of color spaces for ear authentication. *Pattern Recognition*, 42(9), 1906–1913.
- [27] Jayaraman, U., Prakash, S., Gupta, P. (2008). Indexing multimodal biometric databases using Kd-tree with feature level fusion. In *LNCS: Vol. 5352. Proc. of 4th int'l conference on information systems security (ICISS'08)*, Hyderabad, India (pp. 221–234).
- [28] R.Osada, T.Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Transactions on Graphics*, 21(4):807–832, 2002.