

Università degli Studi di Padova  
Dipartimento di Scienze Statistiche  
Corso di Laurea Triennale in  
Statistica per le Tecnologie e le Scienze



RELAZIONE FINALE  
**Dati biometrici: gestione ed ottimizzazione del reperimento**

Relatore Prof. Massimo Melucci  
Dipartimento di Ingegneria dell'Informazione

Laureando: Davide Bettella  
Matricola N. 2003006

Anno Accademico 23/24

# Indice

1. Introduzione
2. Caratteristiche biometriche e tecniche di rilevazione
  - 2.1. Volto
  - 2.2. Impronta digitale
  - 2.3. Iride
  - 2.4. Mano
  - 2.5. Retina
  - 2.6. Dinamica della firma
3. Ricerca di immagini biometriche simili
  - 3.1. Architettura
  - 3.2. Formalizzazione
  - 3.3. Descrittori d'immagine
    - 3.3.1. Descrittori di colore
    - 3.3.2. Descrittori di texture
    - 3.3.3. Descrittori di forma
  - 3.4. Calcolo della similarità tramite Distanza Euclidea
  - 3.5. Efficacia
  - 3.6. Efficienza
4. Miglioramento dell'efficienza della ricerca
  - 4.1. Algoritmo Binning-Pruning
    - 4.1.1. Classificazione di Henry
    - 4.1.2. L'algoritmo
  - 4.2. Filtraggio con proiezioni successive
5. Conclusioni
6. Bibliografia

## **1. Introduzione**

Con il termine “biometria” si fa riferimento alla disciplina che si occupa di misurare ed analizzare tratti fisici e comportamentali degli esseri umani.

Tali misurazioni trovano largo utilizzo nel campo della verifica dell'identità e della previsione basata sulle caratteristiche che un soggetto presenta e, con lo svilupparsi delle tecnologie, l'interesse soprattutto a fini securitari verso l'utilizzo di tali rilevazioni è cresciuto, producendo conseguentemente delle grandi moli di dati biometrici.

Questo elaborato si propone quindi di illustrare le caratteristiche biometriche proposte negli anni e le relative tecnologie di rilevazione, per poi approfondire quelle che compongono la maggior parte delle collezioni di dati biometrici, analizzando gli approcci sviluppati per il riconoscimento di tali dati ed ottimizzare la ricerca in collezioni estese.

## 2. Caratteristiche biometriche e tecniche di rilevazione

Una caratteristica biometrica desiderabile dovrebbe possedere le seguenti caratteristiche [1]:

- Universalità: ogni soggetto dovrebbe presentare la caratteristica.
- Permanenza: la caratteristica dovrebbe restare invariata nel tempo.
- Unicità: ogni soggetto dovrebbe esprimere in maniera unica la caratteristica biometrica.
- Semplicità di rilevazione: la caratteristica dovrebbe poter essere rilevata in maniera rapida, non dispendiosa e non invasiva.

La maggior parte delle caratteristiche biometriche utilizzate dai sistemi di verifica di identità e nelle analisi predittive sono caratteristiche biometriche fisiologiche, preferibili a quelle comportamentali data la maggior semplicità di rilevazione e la maggior affidabilità dei dati prodotti.

Le tecniche di rilevazione delle caratteristiche fisiologiche sono varie, ma principalmente si basano sull'ottica.

Di seguito si mettono in luce le caratteristiche biometriche fisiologiche più comuni e le relative tecniche di rilevazione [2,3].

### 2.1. Volto

Le caratteristiche facciali sono quelle che vengono utilizzate maggiormente dagli esseri umani per riconoscersi tra loro, probabilmente per questo sono anche quelle che dispongono di più dati a supporto degli studi.

Le tecnologie in quest'ambito sono ottiche, si basano o sulla semplice fotografia o sulla mappatura delle caratteristiche facciali come, ad esempio, la distanza tra gli occhi o l'ampiezza del naso. Vengono identificati dei nodi comuni ad ogni faccia e vengono misurate le varie distanze tra essi.

I due metodi principali sono la rilevazione in due dimensioni ed in tre dimensioni: la rilevazione in due dimensioni, a scapito della semplicità di rilevazione, risente delle

condizioni di luminosità e dell'inclinazione del volto; la rilevazione in tre dimensioni utilizza anche le curve dei tessuti rigidi del volto, migliorando così la precisione e la gestione delle problematiche legate alla luminosità, causando però un aumento del costo computazionale.

## 2.2. Impronta digitale

La prima caratteristica biometrica ad essere stata utilizzata nell'ambito del riconoscimento è stata l'impronta digitale. Grazie alla sua immutabilità nel tempo, salvo modifiche causate da traumi, ed alla sua unicità, si pensi al fatto che pure i gemelli omozigoti presentano impronte diverse, ha trovato largo utilizzo nel campo della sicurezza e delle scienze forensi.

La rilevazione della caratteristica si basa sulla rilevazione delle strutture dell'impronta, come creste e solchi, tramite dispositivi ottici, termici, a ultrasuoni o che rilevano la capacità dielettrica.

## 2.3. Iride

Il riconoscimento dell'iride si basa sulle caratteristiche dell'area anulare colorata che circonda la pupilla. Tali caratteristiche si manifestano in modo unico per ogni iride, anche nello stesso soggetto le iridi differiscono per l'occhio destro e sinistro. Il riconoscimento è effettuato tramite acquisizione ottica dell'occhio e sull'immagine acquisita viene localizzata e opportunamente codificata l'area corrispondente all'iride.

L'iride se utilizzata nell'ambito dell'autenticazione presenta un'elevata accuratezza data la sua unicità e ha il pregio di restare invariata nel tempo, ma l'acquisizione dell'immagine risulta difficoltosa: le dimensioni dell'iride sono piccole in confronto ad altre caratteristiche, quindi è necessario l'utilizzo di strumenti ad alta definizione. Inoltre le dimensioni dell'area anulare si modificano in base alle variazioni dell'illuminazione.

## 2.4. Mano

L'utilizzo della mano come caratteristica biometrica si divide in due approcci: l'utilizzo della geometria complessiva della mano e l'impronta del palmo.

### 2.4.1. Geometria della mano

L'acquisizione della caratteristica avviene tramite dispositivi ottici e si basa sulla misurazione di varie caratteristiche quali le dimensioni del palmo, lunghezza ed ampiezza delle dita ed il perimetro della mano.

#### 2.4.2. Impronta del palmo

L'acquisizione dell'impronta del palmo consiste nell'ottenere la disposizione delle strutture di contatto del palmo, che sono più sporgenti rispetto alle altre. Può essere considerata un'estensione dell'acquisizione dell'impronta digitale e le tecnologie di rilevazione sono le stesse: ottiche, termiche, capacitive e ad ultrasuoni.

#### 2.5. Retina

La retina viene utilizzata come caratteristica biometrica grazie all'unicità del pattern di vasi sanguigni presenti in essa. Essendo un tessuto interno all'occhio essa è meno soggetta a modifiche nel tempo rispetto ad altre caratteristiche biometriche, ma si rende necessario evidenziarla tramite illuminazione infrarossa. La diversa capacità di assorbimento di questa fonte luminosa da parte dei vasi sanguigni rispetto ad altri tessuti permette una semplice evidenziazione degli stessi.

#### 2.6. Dinamica della firma

L'utilizzo della firma come caratteristica biometrica si è evoluto dalla semplice comparazione grafica della firma allo studio della dinamica del tratto. La ragione è che una firma è facilmente riproducibile, ma il modo in cui viene tracciata è pressoché impossibile da copiare.

La rilevazione avviene tramite un tablet e consiste quindi nell'acquisire non solo la grafica, ma anche alcune caratteristiche definite comportamentali, quali il movimento sul piano, la pressione applicata, il tempo impiegato, la velocità e l'inclinazione.

Il vantaggio nell'ambito dell'autenticazione è quindi la grande difficoltà di falsificazione, ma presenta un elevato tasso di errore data la dipendenza da molte caratteristiche comportamentali che possono variare spesso in base allo stato psicofisico del soggetto.

### 3. Ricerca di immagini biometriche simili

Le caratteristiche biometriche rilevabili da un essere umano sono molteplici e molto varie tra loro, nonostante ciò, l'essere umano si affida principalmente alla vista per riconoscere altre persone e la caratteristica più utilizzata in natura per tale operazione è il volto. Logica conseguenza è che lo sforzo verso lo studio di questa particolare caratteristica sia stato negli anni particolarmente intenso. Attualmente, quindi, la maggior parte delle collezioni di dati biometrici sono formati da immagini, e conseguentemente lo studio di sistemi per la ricerca di immagini tra loro simili è diventato particolarmente rilevante per le organizzazioni con fini di sicurezza. Si pensi ad esempio alle forze dell'ordine di un Paese come gli Stati Uniti, le quali dispongono di una collezione di decine di milioni di volti e hanno necessità di condurre una ricerca sul territorio nazionale basandosi sulle acquisizioni fotografiche di un ricercato; è impensabile un sistema non informatizzato per la ricerca in tale collezione.

Nei sistemi per tale ricerca, le immagini vengono processate automaticamente da algoritmi che puntano ad estrarre dei vettori di features, i quali rappresentano alcune caratteristiche dell'immagine processata, come colori, forma o texture, tali vettori vengono poi utilizzati per calcolare la similarità tra le immagini nella collezione e l'immagine utilizzata come query.

#### 3.1. Architettura

Il sistema di ricerca svolge due attività principali: inserimento dati e query processing. Si compone di tre elementi, l'interfaccia utente, il modulo per il query processing e la collezione.

L'inserimento dati consiste nell'estrarre i vettori di features dalle immagini ed inserirli nella collezione, il query processing consiste nell'estrarre dalla query i vettori di features, calcolare la similarità con le immagini nella collezione e restituire le immagini più simili alla query. Viene utilizzato per entrambe le attività il modulo di query processing. (Fig. 1.)

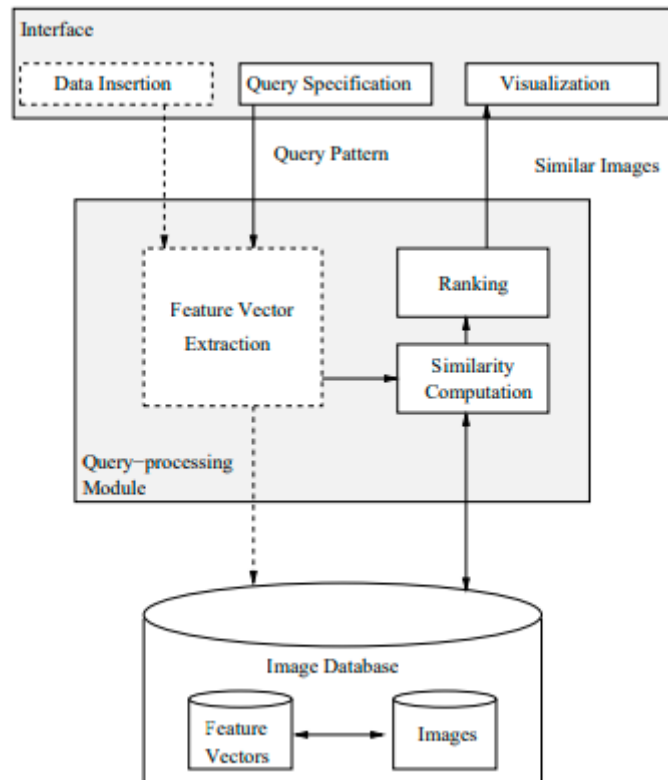


Figura 1 [4]

### 3.2. Formalizzazione

Il sistema di ricerca basa il suo funzionamento sulla costruzione di descrittori d'immagine, oggetti formati da un algoritmo di estrazione che codifica le immagini in vettori di features e da una misura di similarità che confronta le due immagini.

Si definisce immagine  $\hat{I}$  la coppia  $(D_i, \vec{I})$ :

- $D_i$  è un insieme finito di pixel, quindi  $D_i \subset \mathbb{N}^2$
- $\vec{I}: D_i \rightarrow \mathbb{R}^n$  è una funzione che assegna ad ogni pixel  $p$  in  $D_i$  un vettore  $\vec{I}(p) \in \mathbb{R}^n$

Il vettore  $\vec{v}_{\hat{I}}$  delle features è assimilabile ad un punto nello spazio  $\mathbb{R}^n$ ,  $\vec{v}_{\hat{I}} = (v_1, v_2, \dots, v_n)$  con  $n$  dimensione del vettore.

Il descrittore d'immagine  $D$  si definisce come una tupla  $(\varepsilon_D, \delta_D)$  dove:

- $\varepsilon_D: \{\hat{I}\} \rightarrow \mathbb{R}^n$  è una funzione per estrarre il vettore delle features da un'immagine
- $\delta_D: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  è una funzione che calcola la similarità tra due immagini utilizzando la distanza tra i due vettori di features



Di seguito (Figura 2) si mostra la rappresentazione grafica di un ipotetico descrittore  $D$  per la similarità tra due immagini  $\hat{I}_A$  e  $\hat{I}_B$ . L'algoritmo  $\varepsilon_D$  estrae i due vettori di features  $\vec{v}_{\hat{I}_A}$  e  $\vec{v}_{\hat{I}_B}$ . La funzione  $\delta_D$  calcola la similarità  $d$  tra le due immagini.

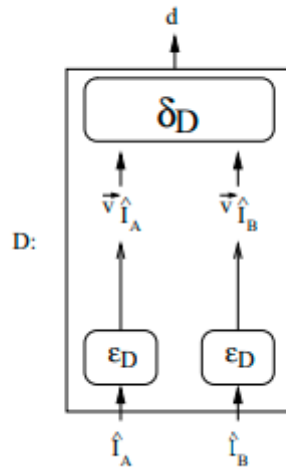


Figura 2 [4]

### 3.3. Descrittori d'immagine

Come precedentemente descritto, i descrittori d'immagine sono delle tuple formate da una funzione di estrazione di features e da una funzione di distanza, usati per calcolare la similarità di due immagini. Ne esistono svariate tipologie:

#### 3.3.1. Descrittori di colore

Il colore di un'immagine è una delle caratteristiche più utilizzate per la ricerca di immagini. Esistono due approcci principali: descrittori che includono informazioni riguardo alla distribuzione spaziale dei colori e descrittori che non le includono.

Un descrittore che ad esempio non comprende la distribuzione spaziale dei colori è l'istogramma dei colori [5], il più utilizzato nel riconoscimento. L'algoritmo di estrazione dell'istogramma si divide in tre parti: divisione degli spazi in aree di colore, associazione delle aree a un determinato comparto dell'istogramma e calcolo del numero di pixel di ogni area e quindi in ogni comparto dell'istogramma. La somiglianza tra due istogrammi viene valutata tramite la distanza Euclidea o tramite l'intersezione tra gli istogrammi stessi.

L'intersezione viene calcolata come segue: dati due istogrammi  $I$  ed  $M$  contenenti ciascuno  $n$  comparti, l'intersezione è definita come

$$\sum_{j=1}^n \min(I_j, M_j)$$

Il risultato è quindi il numero di pixel dello stesso colore in comune per entrambe le immagini.

Un altro esempio di descrittore privo di informazioni spaziali sono i Momenti [4], rispettivamente di primo, secondo e terzo ordine.

Vengono utilizzati per formare il vettore delle features e sono costruiti come segue:

$$E_i = (1/N) \sum_{j=1}^N p_{ij}$$

$$\sigma_i = \sqrt{\left(\frac{1}{N}\right) \sum_{j=1}^n (p_{ij} - E_i)^2}$$

$$s_i = \sqrt[3]{\left(\frac{1}{N}\right) \sum_{j=1}^n (p_{ij} - E_i)^3}$$

Con  $p_{ij}$  valore dell' $i$ -esima componente di colore del pixel  $j$ , e  $N$  il totale di pixel nell'immagine.

Esempi di descrittori che invece considerano la distribuzione spaziale possono essere i vettori di coerenza del colore (CCV), la classificazione dei pixel (BIC) ed il correlogramma di colore.

I vettori di coerenza del colore [6] si calcolano contando per ogni colore il numero di pixels coerenti ( $\alpha_i$ ) ed incoerenti ( $\beta_i$ ), un pixel è definito coerente se il numero delle sue componenti adiacenti tra loro supera una determinata soglia  $\tau$ . Per ogni  $i$ -esimo colore quindi si avrà la coppia ( $\alpha_i, \beta_i$ ). Il vettore di coerenza per l'immagine sarà

$$V_C = \langle (\alpha_1, \beta_1)(\alpha_2, \beta_2), \dots, (\alpha_n, \beta_n) \rangle$$

Con  $n$  numero totale di colori nell'immagine.

Per confrontare due immagini  $I, I'$  con i rispettivi vettori di coerenza

$$V_I = \langle (\alpha_1, \beta_1)(\alpha_2, \beta_2), \dots, (\alpha_n, \beta_n) \rangle$$

$$V_{I'} = \langle (\alpha'_1, \beta'_1)(\alpha'_2, \beta'_2), \dots, (\alpha'_n, \beta'_n) \rangle$$

Non viene utilizzato l'istogramma, poiché come visto sopra considererebbe le quantità totali di pixel per ogni colore, andrebbe quindi a confrontare le quantità  $\alpha_i + \beta_i$  con  $\alpha'_i + \beta'_i$  andando ad ignorare il numero di pixel coerenti.

La misura calcolata per la somiglianza tra due vettori di coerenza, quindi, andrà a confrontare il numero di pixel coerenti e incoerenti per ogni colore:

$$\Delta G = \sum_{j=1}^n |(\alpha_j - \alpha'_j) + (\beta_j - \beta'_j)|$$

Il correlogramma di colore [7] si basa invece sulla correlazione spaziale dei colori nell'immagine. Dati  $p_i$  pixel dell'immagine  $I$ , sia  $I_{c_i}$  l'insieme di pixel dell' $i$ -esimo colore il correlogramma è definito come segue:

$$\gamma_{c_i, c_j}^{(k)} = P_{p_1 \in I_{c_i}, p_2 \in I_{c_j} \mid |p_1 - p_2| = k}$$

In pratica calcola la probabilità che dato un pixel  $p_1$  di colore  $c_i$ , il pixel  $p_2$  che dista  $k$  dal primo, sia di colore  $c_j$ .

Segue la definizione di autocorrelogramma  $\gamma_{c_i, c_i}^{(k)}$  che spiega la correlazione di colori uguali nello spazio.

La classificazione dei pixel [4] consiste nel classificare i singoli pixel come esterni o interni. Un pixel è classificato come esterno se si trova sul bordo dell'immagine o se almeno uno dei quattro pixel adiacenti (superiore, inferiore, destro, sinistro) è di colore differente, viene classificato come interno ogni pixel che ha tutti e quattro i pixel adiacenti dello stesso colore.

Una volta classificati tutti i pixel, si noti che è una classificazione mutualmente esclusiva, si creano due istogrammi differenti e si procede utilizzandoli separatamente.

### 3.3.2. Descrittori di Texture

La texture può essere definita come la distribuzione di alcuni pattern visivi nell'immagine come granularità, direzionalità e ripetitività. Gli approcci per la rappresentazione delle textures sono raggruppati in due classi principali, quelli basati sulla distribuzione spaziale e quelli basati sulla frequenza.

Il più comune tra quelli spaziali è la matrice di co-occorrenza [4] che si propone di estrarre la relazione tra i livelli della componente grigia dei pixels in relazione alla loro posizione.

Tra i descrittori basati sulla frequenza c'è l'algoritmo a ondicelle di Gabor [8], utilizzato nel processare le immagini, come per compressione ricostruzione o analisi. Ampiamente utilizzato nell'analisi dei segnali, il filtro di Gabor serve ad osservare l'energia di un segnale ad una certa scala e con un certo orientamento. Scala ed orientamento sono le features che vengono utilizzate nell'ambito del reperimento basato su texture.

Una funzione bidimensionale di Gabor consiste in un'onda sinusoidale con una certa frequenza ed orientamento, modulata da una Gaussiana bi-dimensionale. Il filtro di Gabor è definito come segue:

$$g, \lambda, \theta, \Psi, \sigma, \gamma (x, y) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \Psi\right)$$

Con

$$\begin{aligned}x' &= x \cos(\theta) + y \sin(\theta) \\y' &= y \cos(\theta) - x \sin(\theta)\end{aligned}$$

Dove  $\lambda$  è la lunghezza d'onda del fattore coseno,  $\theta$  rappresenta l'orientamento della normale rispetto alla funzione di Gabor, l'errore di fase è rappresentato da  $\Psi$  e  $\sigma$  è la deviazione standard della Gaussiana. Il risultato dell'applicazione del filtro è un vettore di valori che rappresentano l'energia contenuta nel segnale a differenti scale e orientamenti. Nell'ambito dell'estrazione delle features valori di energia simili rappresentano immagini o aree con textures simili. Il vettore delle features si compone quindi con la media e la deviazione standard del vettore risultante dall'algoritmo.

### 3.3.3. Descrittori di Forma

I descrittori di forma sono particolarmente utili per identificare un'oggetto all'interno di un'immagine.

Si dividono principalmente tra basati sul contorno e sulle aree. Queste due classi si dividono a loro volta in descrittori globali e locali, dipendentemente dal fatto che rappresentino una forma completa o segmentata.

Il primo descrittore si basa sull'invarianza dei momenti [8,9], l'algoritmo d'invarianza dei momenti di Hu è uno dei più utilizzati nell'estrazione di features per il riconoscimento di oggetti. Spesso le immagini, ad esempio nell'ambito della sicurezza, non sono chiare e gli oggetti presenti non sono facilmente visibili né identificabili in termini di misure, collocazione nello spazio e orientamento. L'algoritmo di invarianza dei momenti ha quindi particolare rilevanza nell'ambito del riconoscimento delle forme, poiché invariante rispetto a posizione, misura ed orientamento. L'algoritmo si propone inoltre come automatizzazione della fase di pre-processing che viene svolta manualmente per preparare le immagini al riconoscimento, come allineamento e adattamento della misura dell'immagine, procurando delle features indipendenti da posizione misura e orientamento dell'oggetto nell'immagine.

Si definisce il momento bidimensionale di ordine  $(p + q)$  di un'immagine  $f(x, y)$  come

$$m_{p,q} = \sum_x \sum_y x^p y^q f(x, y)$$

Con  $p, q = 0,1,2$

Il relativo momento centrale è definito come

$$\mu_{p,q} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y)$$

Con  $\bar{x} = m_{1,0}/m_{0,0}$  e  $\bar{y} = m_{0,1}/m_{0,0}$  definiti centro della regione.

I momenti centrali fino al terzo ordine si calcolano come segue

$$\mu_{0,0} = m_{0,0}$$

$$\mu_{1,0} = 0$$

$$\mu_{0,1} = 0$$

$$\mu_{1,1} = m_{1,1} - \bar{y} m_{1,0}$$

$$\mu_{2,0} = m_{2,0} - \bar{x}^2 m_{1,0}$$

$$\mu_{0,2} = m_{0,2} - \bar{y}^2 m_{0,1}$$

$$\mu_{3,0} = m_{3,0} - 3\bar{x} m_{2,0} + 2\bar{x}^2 m_{1,0}$$

$$\mu_{0,3} = m_{0,3} - 3\bar{y} m_{0,2} + 2\bar{y}^2 m_{0,1}$$

$$\mu_{2,1} = m_{2,1} - 2\bar{x} m_{1,1} - \bar{y} m_{2,0} + 2\bar{x}^2 m_{0,1}$$

$$\mu_{1,2} = m_{1,2} - 2\bar{y} m_{1,1} - \bar{x} m_{0,2} + 2\bar{y}^2 m_{1,0}$$

Il momento centrale normalizzato di ordine  $(p + q)$  è definito

$$\eta_{p,q} = \mu_{p,q} / \mu_{0,0}^\gamma$$

Con  $p, q = 0, 1, 2 \dots$  dove  $\gamma = \left\lceil \frac{p+q}{2} \right\rceil + 1$

Quando  $(p + q) = 2, 3, \dots$  si può ricavare dai momenti centrali di secondo e terzo ordine un insieme di momenti descritti come segue

$$\begin{aligned} \phi_1 &= \mu_{2,0} + \mu_{0,2} \\ \phi_2 &= (\mu_{2,0} + \mu_{0,2})^2 + (4\mu_{1,1})^2 \\ \phi_3 &= (\mu_{3,0} + 3\mu_{1,2})^2 + (3\mu_{2,1} - \mu_{0,3})^2 \\ \phi_4 &= (\mu_{3,0} + \mu_{1,2})^2 + (\mu_{2,1} - \mu_{0,3})^2 \\ \phi_5 &= (\mu_{3,0} + 3\mu_{1,2}) + (\mu_{3,0} + \mu_{1,2})[(\mu_{3,0} + \mu_{1,2})^2 - 3(\mu_{2,1} + \mu_{0,3})^2] + (3\mu_{2,1} \\ &\quad + \mu_{0,3})(\mu_{2,1} + \mu_{0,3})[3(\mu_{3,0} + 3\mu_{1,2})^2 + (3\mu_{2,1} - \mu_{0,3})^2] \\ \phi_6 &= (\mu_{2,0} - \mu_{0,2})[(\mu_{3,0} + \mu_{1,2})^2 - (\mu_{2,1} + \mu_{0,3})^2] \\ &\quad + 4\mu_{1,1}(\mu_{3,0} + \mu_{1,2})(\mu_{2,1} + \mu_{0,3}) \\ \phi_7 &= (3\mu_{2,1} - \mu_{0,3})(\mu_{3,0} - \mu_{1,2})[(\mu_{3,0} + \mu_{1,2})^2 - 3(\mu_{2,1} + \mu_{0,3})^2] - (\mu_{3,0} \\ &\quad - 3\mu_{0,3})(\mu_{2,1} + \mu_{0,3})[3(\mu_{3,0} + \mu_{1,2})^2 - (\mu_{2,1} + \mu_{0,3})^2] \end{aligned}$$

L'insieme di momenti così ottenuti hanno delle caratteristiche apprezzabili come descrittori di forma, in particolare l'insieme  $\phi_1 - \phi_6$  è un insieme di momenti invarianti rispetto alla scala, alla rotazione e alla traslazione.  $\phi_7$  serve a differenziare le immagini speculari. Questi momenti vengono quindi usati per popolare il vettore delle features.

Un altro descrittore è l'angolatura del fascio [4,10], il descrittore si basa sui fasci generato dai pixels del contorno, ottenuti dall'insieme di linee rette che collegano ogni singolo pixel ad ogni altro pixel sul contorno della forma. Basandosi sull'angolatura di questi fasci si ottiene un descrittore di forma.

#### 3.4. Calcolo della similarità con distanza euclidea

Come accennato in precedenza il calcolo della similarità tra due immagini è lo step più importante in un sistema per la ricerca di immagini simili. La similarità è una misura calcolata numericamente che rappresenta la forza delle connessioni tra le due immagini. La misura più utilizzata è la distanza Euclidea tra i due vettori di features. Siano  $M^k$  e  $M^t$  due vettori di feature, provenienti dalla query e dalla collezione rispettivamente, la distanza Euclidea è calcolata come segue [8]

$$ED(M^k, M^t) = \sqrt{\sum_{i=1}^n (M_i^k - M_i^t)^2}$$

Dove  $M_i^k$  e  $M_i^t$  sono i valori della feature  $i$ .

Valori inferiori della distanza Euclidea corrispondono ad una somiglianza elevata.

### 3.5. Efficacia

L'esistenza di diversi descrittori d'immagine crea molteplici possibilità di costruzione di un sistema di ricerca. Dipendentemente dal contesto di applicazione e dalle necessità derivanti si vuole eleggere il descrittore più adatto ai propri scopi. Per decidere il descrittore designato è necessario condurre quindi una serie di sperimentazioni per valutare l'efficacia dei diversi descrittori. La valutazione dell'efficacia è un'operazione complicata che necessita la scelta di una collezione di immagini adeguata, un insieme di immagini da utilizzare come query, l'insieme delle immagini rilevanti, nella collezione per ogni query ed infine la scelta della misura di similarità.

Un'immagine biometrica rilevante per una data query è ad esempio nel caso di sistemi di identificazione, un'immagine appartenente allo stesso individuo.

La valutazione viene condotta utilizzando l'insieme delle query e comparando i risultati delle immagini rilevanti forniti dal sistema con quelle decise a priori.

Come visto precedentemente un descrittore rappresenta un'immagine come un punto nello spazio n-dimensionale, l'efficacia del descrittore sarà più elevata in base a quanto saranno separati definitivamente tra loro i cluster di immagini rilevanti, mentre maggiore sarà la compattezza all'interno di ogni cluster, maggiore sarà la robustezza del descrittore al crescere del numero di classi. Da qui segue che una misura dell'efficacia dovrebbe comprendere il concetto di separabilità e di compattabilità. Più formalmente la compattabilità del descrittore indica la sua invarianza rispetto a caratteristiche di oggetti appartenenti alla stessa classe, mentre la separabilità sarebbe la capacità di discriminare tra oggetti di classi differenti.

Nonostante l'ampio uso di questi concetti nella cluster analysis, nella letteratura dei sistemi di riconoscimento di immagini non si nota una particolare attenzione verso di essi. Difatti uno dei metodi più utilizzati è la curva Precision×Recall.

La precisione viene definita come il rapporto le immagini rilevanti reperite e il totale delle immagini reperite, il recall è definito come il rapporto tra le immagini rilevanti reperite e il totale delle immagini rilevanti nella collezione [4].

### 3.6. Efficienza

L'efficienza di un sistema di ricerca di immagini simili in termini di tempo di reperimento è dipendente dalla dimensione della collezione e dei suoi elementi. Come precedentemente detto un'immagine è rappresentata come un punto in uno spazio n-dimensionale, dove n può essere elevato. In questo caso i sistemi che funzionano bene in contesti a dimensionalità ridotta, hanno risultati insoddisfacenti. Un approccio comune per fronteggiare la "maledizione della dimensionalità" consiste nell'applicare metodologie di riduzione della dimensionalità, come l'Analisi delle Componenti Principali.



## 4. Miglioramento dell'efficienza della ricerca

Nel precedente capitolo è stato illustrato il funzionamento dei sistemi di ricerca di immagini simili, per la rilevanza che hanno le immagini all'interno della biometria. Ora, si vuole procedere ad illustrare delle metodologie che permettono il miglioramento dell'efficienza della ricerca di similarità o del pattern matching per caratteristiche biometriche anche diverse da immagini, nonostante queste rappresentino sempre la maggior quantità di dati.

Vengono presentati quindi degli schemi concettuali e algoritmi che prescindono dallo specifico tipo di sistema di riconoscimento, si tenga conto però, che sono sviluppati in ambito di applicazioni di identificazione, in breve dei sistemi che dopo aver calcolato la similarità con la query restituiscono gli elementi che presentano un punteggio di similarità superiore ad una certa soglia, detti "match", e le rispettive identità.

### 4.1. Algoritmo Binning-Pruning

Il metodo qui presentato nasce dalla necessità di affrontare il problema della scalabilità dei comuni sistemi di ricerca di dati biometrici o pattern matching, creati inizialmente per avere grande accuratezza su collezioni relativamente piccole. L'aumento del numero di dati biometrici raccolti, e conseguentemente delle dimensioni delle collezioni, ha reso necessario un approccio differente. Basti pensare al riconoscimento dei volti di persone schedate o altre applicazioni per la sicurezza su vaste popolazioni, qui il tempo e l'efficienza diventano molto più rilevanti, pur non volendo trascurare l'accuratezza. I database tradizionali solitamente possono indicizzare i record su una scala ordinale, alfabetica o numerica, per un reperimento efficiente, questo tipicamente non è possibile quanto si tratta di collezioni biometriche, poiché non esiste un ordine naturale con cui organizzare i vari record. Si pensi alla differenza tra un database di record testuali e una collezione di immagini. Il metodo [11] si propone di guidare la ricerca del sistema in una collezione biometrica classificando in maniera "rozza" la query in input per poi praticare una ricerca più precisa sulla similarità.

La collezione viene suddivisa in gruppi (“bins”) utilizzando l’algoritmo k-means, la query viene anch’essa assegnata ad uno di questi gruppi cosicché tutti gli elementi di quel gruppo siano considerati dei risultati validi per la query. Vengono presi in considerazione anche i C gruppi più vicini, per evitare che i dati più rilevanti, o i match nel caso delle applicazioni identificative, trovandosi magari nei gruppi vicini, vengano ignorati. Chiaramente il numero C dipende dal contesto in cui viene applicato il metodo.

#### 4.1.1. Classificazione di Henry

L’algoritmo prende ispirazione dalla procedura definita come il Sistema di Classificazione di Henry, utilizzato per la classificazione delle impronte digitali. Questo sistema classifica un dato contenente le impronte di tutte le dita di una mano in 1024 gruppi diversi, basandosi sulle minuzie delle impronte, come creste, archi ecc. di ogni dito.

Questo sistema non si dimostra particolarmente efficace a causa della distribuzione disomogenea tra le classi, è stato dimostrato infatti che 2/5 delle classi contenevano il 65% delle impronte. Inoltre, ciò che limita l’utilizzo di questo metodo, è chiaramente la non estendibilità ad altre caratteristiche biometriche.

Da qui si formalizza un metodo più generalizzato per la ricerca nelle collezioni biometriche estese.

#### 4.1.2. L’algoritmo

L’algoritmo si propone di ridurre lo spazio in cui ricercare il match attraverso la partizione della collezione in diversi gruppi.

Utilizzando tale partizionamento si ottiene una divisione della collezione che riunirà nello stesso gruppo i dati biometrici che sono simili tra loro.

In generale è possibile rappresentare un dato biometrico come un vettore  $k$ -dimensionale  $X_i = [x_{i1}, x_{i2}, \dots, x_{ik}]$ , la problematica risiede nel classificare tutti gli  $N$  vettori di dati all’interno della collezione per formare uno spazio vettoriale  $S$  suddiviso in  $M$  classi  $\{Y_1, Y_2, \dots, Y_M\}$ .

Quindi

$$\bigcup_{i=1}^N Y_i = S$$

Dove

$$Y_i \cap Y_j = \emptyset, \forall i \neq j$$

In ognuna di queste classi viene eletto un elemento rappresentativo che verrà utilizzato per il confronto con la query (figura 3).

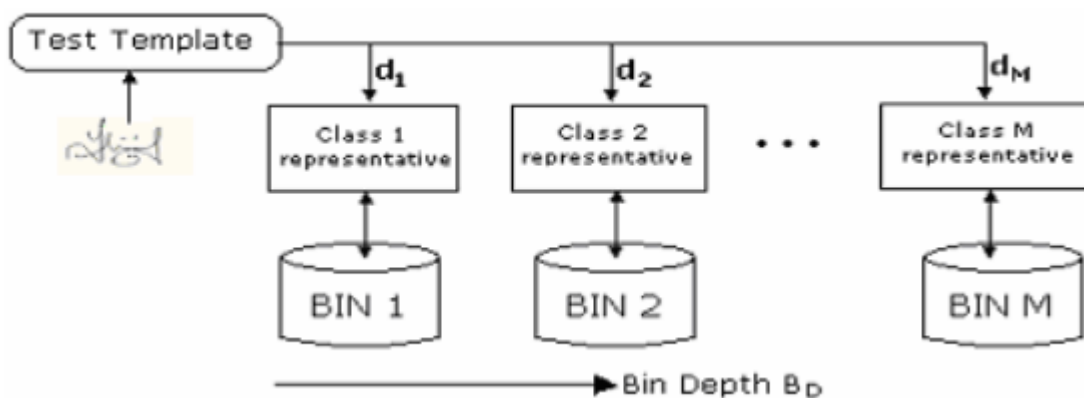


Figura 3 [11]

Nel caso di una comparazione 1:  $N$  per l'identificazione, come in un sistema semplice, il tempo necessario per valutare tutte le similarità tra la query e gli elementi della collezione è chiaramente di  $N$  operazioni. Possiamo quindi definire il costo di questa metodologia di reperimento come  $\Theta(N)$ .

Usando l'approccio suggerito dal binning-pruning, quando si procede ad effettuare la ricerca di una query all'interno della collezione, è necessario identificare i  $C$  gruppi più vicini alla query, all'interno dei quali si dovrebbe trovare l'elemento cercato. Per far ciò vengono calcolate prima le distanze della query dagli elementi rappresentativi di ogni classe, tipicamente i centri, e successivamente vengono confrontati gli elementi all'interno dei gruppi selezionati con la query.

La scelta dei  $C$  gruppi può richiedere nel peggiore dei casi  $C \times M$  comparazioni. Conseguentemente il costo computazionale di questa

operazione sarà  $\Theta(C \times M)$ , portando il costo complessivo dell'identificazione a una complessità

$$\Theta(C \times M) + C \times \Theta(A_M)$$

Dove viene definito  $A_M$  come la cardinalità media di ogni gruppo

$$A_M = \frac{N}{M}$$

Se si considera che  $C < M \ll N$  per come è formulato l'algoritmo e  $A_M < N$  risulta che

$$\Theta(C \times M) + C \times \Theta(A_M) < \Theta(N)$$

Viene quindi giustificato l'utilizzo dell'algoritmo binning-pruning dal punto di vista dell'efficienza, in quanto richiede sensibilmente meno tempo rispetto ad una scansione della collezione con modalità "brute force".

Si noti che utilizzando la notazione sopra si fa riferimento ad una situazione in cui si assume omogenea la distribuzione delle occorrenze all'interno dei gruppi, problematica che riporterebbe al problema della classificazione di Henry. Si risolve facilmente sostituendo all'interno del prodotto  $C \times \Theta(A_M)$  il fattore  $\Theta(A_M)$  con  $\Theta(P_S \times N)$ , dove  $P_S$  rappresenta la frazione maggiore della collezione dopo la partizione. La complessità si mantiene vantaggiosa anche in questo caso, poiché  $P_S \times N < N$ .

#### 4.2. Filtraggio con proiezioni successive

Il metodo [12] qui presentato si avvicina all'ottimizzazione in modo totalmente diverso da quello precedente. Invece di agire sulla struttura macroscopica della collezione si propone di ridurre i costi computazionali a

partire dalla semplificazione del calcolo delle similarità sul singolo dato biometrico.

Come visto nello specifico per i sistemi di ricerca di immagini simili, le caratteristiche biometriche possono essere rappresentate da dei vettori di features, i quali possono raggiungere dimensioni elevate, rendendo così il calcolo delle singole similarità molto oneroso.

L'idea è quella di ridurre i vettori di features a vettori monodimensionali, eseguire così una scansione grossolana di tutti gli elementi della collezione, eleggere alcuni di essi a probabili match ed eseguire un confronto accurato tra la query e quest'ultimo insieme di elementi.

Formalizzando, è necessario definire una funzione  $P_j()$  che mappi il vettore di features  $V(Y)$  della caratteristica  $Y$ .

Si vuole ottenere quindi

$$v_j(Y) = P_j(V(Y))$$

Con  $v_j(Y) \in \mathbb{R}$ .

Si può pensare alla funzione  $P_j()$  come ad una proiezione dallo spazio  $d$ -dimensionale del vettore  $V(Y)$  ad uno spazio monodimensionale di features. La fase di filtraggio, quindi, discrimina tra gli elementi della collezione utilizzando queste proiezioni monodimensionali, rendendo così la fase di confronto con la query estremamente rapida. Per migliorare la precisione nella scelta degli elementi eletti a possibili match la procedura propone un filtraggio a cascata, ossia a fasi ripetute, utilizzando una proiezione differente per ogni fase  $j$ .

Viene utilizzata una proiezione lineare per la sua efficienza e la capacità di rappresentare strutture presenti in spazi ad elevata dimensionalità.

Una proiezione randomica è una rappresentazione debole ma efficiente di un elemento della collezione. Se gli elementi rilevanti si trovano all'interno di una sfera di raggio  $r$  nello spazio  $d$ -dimensionale, si troveranno invece in una finestra di dimensione  $2r$  nella proiezione lineare.

Ad ogni fase vengono scartati tutti gli elementi della collezione la cui proiezione si trova all'esterno della finestra calcolata nell'intorno della query.

In figura 4 si può osservare il risultato di una sequenza di proiezioni nel caso di uno spazio di features bidimensionale.

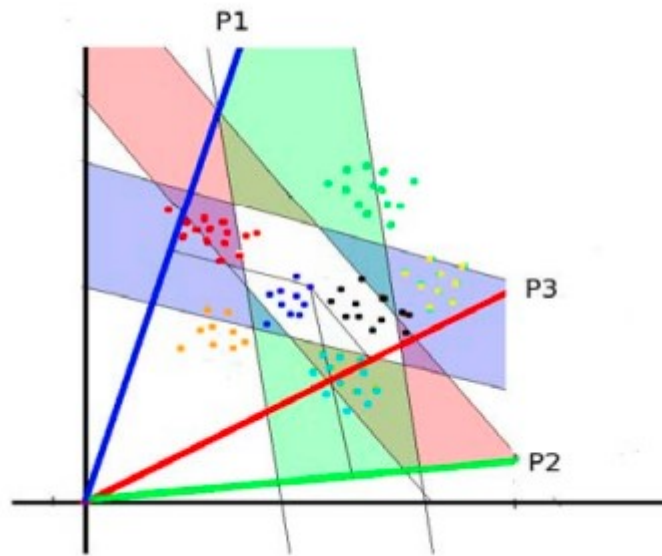


Figura 4: Una sequenza di 3 proiezioni casuali P1, P2, P3, gli elementi al di fuori dell'area bianca al centro delle proiezioni sono rimossi dall'insieme dei possibili candidati. [12]

L'insieme risultante dalla sequenza di filtraggio chiaramente non dipende dall'ordine di applicazione delle proiezioni, ma l'efficienza della procedura sì. Infatti, se le proiezioni iniziali rimuovono un grande numero di elementi della collezione, il numero di confronti nelle fasi finali si riduce considerevolmente. La quantità che si vuole massimizzare nella scelta delle proiezioni è la seguente

$$Score_i = \frac{c_i}{1 + f_i}$$

Con

$$c_i = \frac{\sum_{j \notin W} not S(j)}{\sum_j not S(j)}; f_i = \frac{\sum_{j \in W} S(j)}{\sum_j S(j)}$$

Dove  $S(j)$  è una variabile indicatrice che ha valore 1 quando l'elemento  $j$  è rilevante per la query, mentre l'insieme  $W$  è quello selezionato dalla finestra identificata con le proiezioni successive. La dimensione della finestra di ogni proiezione viene calcolata sull'insieme di training, ed è decisa in modo tale

che ogni elemento rilevante non venga tagliato fuori. Si noti che lo score viene calcolato ad ogni fase dopo il filtraggio della collezione delle fasi precedenti, quindi dopo l'eliminazione di alcuni elementi.

L'algoritmo opera nel seguente modo:

### Algoritmo 1: fase di Learning

```
for i <- numero di proiezioni:
  for ogni possibile proiezione  $P_j$ :
    if la proiezione non è ancora selezionata:
      calcola lo score
      seleziona la proiezione con lo score maggiore e
      rimuovi gli elementi esterni alla finestra
      selezionata
return l'ordine delle proiezioni
```

Per ogni query  $Q$ , viene determinato  $v_j(Q)$  per la  $j$ -esima fase. Vengono rimossi tutti gli elementi che si trovano al di fuori della finestra  $W_j$  calcolata in fase di training.

La porzione della collezione rimasta dopo la sequenza di proiezioni viene restituita per il calcolo delle similarità. L'algoritmo 2 mostra il processo di scelta dei candidati.

### Algoritmo 2: Calcolo dei candidati

```
ListaCandidati <- {Tutti gli elementi della
collezione}
```

```
for ogni proiezione  $P_j$ :
```

```
  Reperisci i valori delle proiezioni degli elementi
  di ListaCandidati per  $P_j$ 
```

Individua la finestra attorno alla proiezione della query

Rimuovi gli elementi di ListaCandidati esterni alla finestra

**return** ListaCandidati

L'efficacia e l'efficienza sono misurate spesso con due parametri: il tasso di successo e il tasso di penetrazione. Il tasso di successo è definito come la probabilità che venga reperita l'identità corretta, il tasso di penetrazione è definito dalla frazione di identità reperite ad ogni query.

Dato che ad ogni fase viene rimosso un certo numero di elementi della collezione, c'è la possibilità che essi fossero rilevanti, si dichiara un fallimento se dopo il filtraggio l'insieme di candidati non presenta nessun elemento rilevante per la query. Dopo ogni fase il tasso di fallimento tende ad aumentare, e tende a diminuire il tasso di penetrazione. Si nota però che nelle prime fasi di filtraggio risulta una grande diminuzione del tasso di penetrazione, ma un tasso di successo che si mantiene vicino al 100%.

Procedendo la velocità con cui diminuisce il tasso di penetrazione si riduce, e si riduce sensibilmente anche il tasso di successo

Le sperimentazioni su questo approccio dimostrano una considerevole riduzione nel carico computazionale, rendendolo valido per approcci di ricerca su collezioni estese.



## **5. Conclusioni**

I sistemi di ricerca di dati biometrici simili applicati a collezioni che si fanno sempre più vaste, si devono necessariamente confrontare con un carico computazionale sempre più elevato. I metodi per affrontare questa problematica sono molteplici e variano nell'approccio: riducono il numero di comparazioni, riducono il costo del singolo confronto o ottimizzano le implementazioni software e hardware [13].

In questo elaborato sono stati presentati due diversi metodi che utilizzano i primi due approcci rispettivamente.

Con l'aumentare delle collezioni è ragionevole aspettarsi che la ricerca nel campo della riduzione del carico computazionale di questi sistemi riceva sempre maggior interesse.

## 6. Bibliografia

- [1] Schmid, N.A., and J.A. O’Sullivan. “Performance Prediction Methodology for Biometric Systems Using a Large Deviations Approach.” *IEEE Transactions on Signal Processing*, vol. 52, no. 10, Oct. 2004, pp. 3036–3045
- [2] Gaur, Shweta, et al. “Biometric Recognition Techniques: A Review.” *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Energy*, vol. 1, no. 4, 1 Jan. 2012, pp. 282–290.
- [3] Choudhary, Jitendra. “Survey of Different Biometrics Techniques.” *International Journal of Modern Engineering Research*, vol. 2, no. 5, Oct. 2012, pp. 3150–3155.
- [4] da Silva Torres, Ricardo, and Alexandre X Falcão. “Content-Based Image Retrieval: Theory and Applications.” *Revista de Informática Teórica E Aplicada*, vol. 13, no. 2, 1 Jan. 2006, pp. 161–185.
- [5] J. Swain, Michael, and Dana H. Ballard. “Color Indexing.” *International Journal of Computer Vision*, vol. 7, no. 1, 1991, pp. 11–32.
- [6] Pass, Greg, et al. “Comparing Images Using Color Coherence Vectors.” *Proceedings of the Fourth ACM International Conference on Multimedia*, 1997.
- [7] Huang, Jing, et al. “Image Indexing Using Color Correlograms.” *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, U.S. Department of Energy Office of Scientific and Technical Information, 17 June 1997.
- [8] Iqbal, Kashif, et al. “Content-Based Image Retrieval Approach for Biometric Security Using Colour, Texture and Shape Features Controlled by Fuzzy Heuristics.” *Journal of Computer and System Sciences*, vol. 78, no. 4, July 2012, pp. 1258–1277.
- [9] Ming-Kuei Hu. “Visual Pattern Recognition by Moment Invariants.” *IEEE Transactions on Information Theory*, vol. 8, no. 2, Feb. 1962, pp. 179–187.

- [10] Arica, Nafiz, and Fatos T. Yarman Vural. "BAS: A Perceptual Shape Descriptor Based on the Beam Angle Statistics." *Pattern Recognition Letters*, vol. 24, no. 9-10, June 2003, pp. 1627–1639.
- [11] Amit Mhatre, et al. "Efficient Search and Retrieval in Biometric Databases." *Proceedings of SPIE*, SPIE, 28 Mar. 2005.
- [12] Iqbal, Atif, and Anoop Namboodiri. "Cascaded Filtering for Fingerprint Identification Using Random Projections." *Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 1 June 2012.
- [13] Drozdowski, Pawel, Christian Rathgeb, and Christoph Busch. "Computational workload in biometric identification systems: An overview." *IET Biometrics* 8.6 (2019): 351-368.