

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

---

---

SCUOLA DI SCIENZE  
Corso di Laurea in Matematica

**APPLICAZIONI  
DELL'OMOLOGIA PERSISTENTE  
AL RICONOSCIMENTO  
DI ESPRESSIONI FACCIALI**

**Relatore:**  
**Prof. MASSIMO FERRI**

**Presentata da:**  
**ELISA BOTTEGHI**

**Anno Accademico 2018/2019**



*Ma non avere paura,  
non ti fermare mai,  
perché il mio amore è fedele  
e non finisce mai.*

(Claudio Chieffo, Non avere paura)

*Alla mia famiglia*



# Indice

<b>Introduzione</b>	<b>iii</b>
<b>1 Utilizzi del riconoscimento delle espressioni facciali</b>	<b>1</b>
<b>2 Omologia</b>	<b>5</b>
2.1 Omologia simpliciale . . . . .	7
2.2 Omologia persistente . . . . .	10
2.3 Barcode e diagramma di persistenza . . . . .	12
2.4 Entropia persistente . . . . .	14
<b>3 Applicazioni</b>	<b>15</b>
3.1 Triangolazione di Delaunay . . . . .	15
3.2 Complesso di Vietoris-Rips . . . . .	18
3.3 Funzioni filtranti . . . . .	20
3.4 Esperimenti . . . . .	22
3.5 Altre ricerche . . . . .	25
<b>Conclusioni</b>	<b>29</b>
<b>Bibliografia</b>	<b>32</b>



# Introduzione

Il riconoscimento e la classificazione di espressioni facciali sono un'area di ricerca molto attiva. Da esse traspaiono condizioni salute, emozioni ed altre informazioni utili in svariati ambiti, dalla medicina all'elaborazione del feedback sulla soddisfazione dei clienti nei servizi. La difficoltà principale è il riconoscimento della forma, dato che non si può avere un volto come standard uguale per tutti. La topologia persistente è una tecnica molto recente in grado di fornire strumenti adeguati per la risoluzione del problema.

Questo elaborato analizza gli studi sull'uso dell'omologia persistente per il problema del riconoscimento di espressioni facciali eseguiti dall'Accademia Cinese delle Scienze [8], l'Università del Lancashire Centrale, il Consiglio Nazionale delle Ricerche dell'Italia, l'Università di Linköping, l'Istituto di Informatica, la Fondazione per la Ricerca e la Tecnologia, l'Università Norvegese della Scienza e della Tecnologia [12].

Nel primo capitolo spieghiamo perché è importante studiare le espressioni e descriviamo alcuni utilizzi.

Nel secondo capitolo presentiamo le definizioni salienti della teoria dell'omologia persistente, che verrà utilizzata per l'analisi dell'espressione. In particolare definiremo il barcode e il diagramma di persistenza.

Nel terzo capitolo mostriamo il metodo attuale per analizzare dati e riportiamo i risultati principali ottenuti con l'omologia persistente.





# Capitolo 1

## Utilizzi del riconoscimento delle espressioni facciali

Negli ultimi anni la tecnologia è diventata fondamentale nella vita di ognuno, ormai tutti hanno un cellulare e si può usufruire di applicazioni di ogni genere per migliorare lo stile di vita. Un aspetto molto importante nella nostra vita sono le emozioni, che vengono trasmesse esternamente dal linguaggio del corpo, in particolare dalle espressioni facciali e dalla voce. Man mano che le macchine diventano sempre più sofisticate e parte integrante della vita quotidiana si fa sempre più interessante studiare come poter insegnare ad una macchina a riconoscere non solo gli aspetti biologici di un essere umano ma anche quelli emotivi. È stato scoperto che un computer in grado di comprendere le emozioni svolge meglio il suo compito nell'assistere un individuo [3]. La percezione accresciuta delle emozioni umane potrebbe permettere ad un robot, per esempio, di evitare azioni che porterebbero a peggiorare lo stato emotivo dell'uomo e addirittura di influenzarlo ad agire nel modo migliore per il suo benessere. Inoltre si potrebbe programmare un'applicazione che rilevi l'emozione dell'utente tramite la fotocamera dello smartphone e raccomandi strategie per affrontare le emozioni negative e migliorare l'umore dell'utente [4]. La tecnologia che prende in considerazione anche le emozioni prende il nome di *affective computing*. Essa si applica in

diversi ambiti, specialmente nella medicina.

È il caso esposto nell'articolo [5], in cui viene stimato il livello di coinvolgimento di alcuni bambini autistici tramite le immagini del viso. Sono stati utilizzati dei dati video di 30 bambini con ASC (Autism Spectrum Condition) aventi background culturali diversi registrati durante una singola sessione di terapia dell'autismo assistita dai robot. ASC è una complessa condizione di sviluppo neurologico caratterizzata da sfide comunicative socio-emotive, nonché comportamenti ripetitivi e stereotipati. Alcune sfide sociali derivano dal fatto che le loro capacità motorie e le loro espressioni, verbali e non, sono molto differenti da quelle comuni. Le loro manifestazioni personali di coinvolgimento inoltre possono variare ampiamente tra i bambini e di solito sono percepite come di bassa intensità, rispetto a quelle dei loro coetanei neurotipici. Questo rende il riconoscimento del coinvolgimento estremamente difficile da eseguire. Un altro aspetto importante delle manifestazioni di coinvolgimento percepito sono le differenze culturali tra i bambini con ASC; esse possono fornire una migliore comprensione della percezione dei sintomi e dell'espressione di comportamenti diversi. Negli studi precedenti all'articolo in esame si è cercato di automatizzare la misurazione del coinvolgimento analizzando le espressioni facciali, la voce e i movimenti del corpo. Il *deep learning* è stato applicato ai bambini con sindrome dell'autismo controllando le immagini del cervello, mentre la stima dell'affetto e del coinvolgimento è stata calcolata sulla stima di segnali comportamentali diversi, come il volto, la posa della testa e la voce. Si indaga per la prima volta in questo articolo l'uso del *deep learning* per misurare in maniera automatizzata i livelli di coinvolgimento di bambini con ASC direttamente a partire dalle immagini dei volti.

Nell'articolo [6] viene proposto un metodo personalizzato per stimare il dolore di una persona che si basa sulle espressioni del volto. Il dolore è un'esperienza molto soggettiva e una misurazione accurata di diverse variabili fisiologiche non è sufficiente, per cui sono state sviluppate varie scale di valutazione della sua intensità basate sul giudizio del paziente stesso. Una scala

analogica molto utilizzata è VAS (Visual Analogue Scale), in cui si chiede al paziente di contrassegnare la sua sofferenza in una scala da 0 (che equivale a non percepire alcun dolore) a 10 (peggior dolore di sempre). Sebbene queste spesso siano utili, i sistemi automatici di rilevazione del dolore possono ridurre gli sforzi di acquisizione del punteggio di esso negli studi su larga scala stimandolo direttamente dalle espressioni facciali dei partecipanti. Il sistema automatico di rilevazione del dolore proposto è basato su VAS e si differenzia dai precedenti metodi automatici perché tiene conto del fatto che la risposta facciale del dolore dipende da numerosi fattori e la relazione tra i segnali del volto e VAS non è chiara. Infatti molti studi hanno trovato basse correlazioni tra espressione facciale e male autoriferito ed è possibile che due persone mostrino diversi livelli di espressione del dolore per lo stesso stimolo. Questo è il primo approccio personalizzato per stimare automaticamente la scala VAS dalle immagini dei volti.

L'*affective computing* è importante non solo nel campo della medicina ma anche in altri ambiti. La musica e le emozioni sono indissolubilmente legate e il rapporto tra le due ha affascinato gli uomini per secoli. Nel documento [7] viene spiegata la progettazione e l'implementazione di un sistema di generazione e riproduzione musicale, Adaptive Music for Affect Improvement (AMAI), il cui obiettivo finale è di migliorare lo stato d'animo dell'ascoltatore. Utilizza delle tecniche per adattare elementi della musica ascoltata e tali adattamenti sono fatti in risposta ai livelli di valenza dell'ascoltatore misurati attraverso l'espressione facciale e il rilevamento delle emozioni.

Date le numerose applicazioni sono stati sviluppati diversi modelli matematici. Esistono modelli che per semplicità non prendono in considerazione le differenze tra le persone di un gruppo ma questi producono risultati meno accurati quando si considera un gruppo di persone provenienti da culture differenti. La personalizzazione di un modello sfrutta i dati specifici di una persona con l'obiettivo di adattare un classificatore generale ai singoli individui. Ciò ha mostrato notevoli miglioramenti nelle attività di apprendimento automatico relative all'analisi dei dati umani. Nell'articolo [4] si è studiato

un tipo specifico di personalizzazione del modello per la stima dell'affetto facciale (valenza e eccitazione).

In questa tesi esaminiamo lavori che propongono l'omologia persistente come metodo matematico per l'analisi di espressioni facciali.

# Capitolo 2

## Omologia

In questo capitolo definiamo i principali concetti dell'omologia persistente, che sarà lo strumento principale di questa tesi per il riconoscimento di espressioni facciali.

Individuare oggetti diversi ma con determinate caratteristiche in comune e considerarli in un'unica classe è molto importante in matematica. Ad esempio due uomini uno in piedi e l'altro seduto sono considerati allo stesso modo se si esamina la specie, mentre altre volte può essere importante la posizione ed un uomo seduto si considera più simile ad un animale seduto. Nel caso di oggetti tridimensionali la **geometria** offre ottimi strumenti: grazie a movimenti rigidi possiamo sovrapporre una figura ad un'altra. Tuttavia è molto rigida e questa caratteristica crea problemi in un contesto in cui le immagini sono di origine naturale, come le espressioni. Allora si potrebbe pensare di utilizzare la **topologia** che associa ad una stessa classe due figure  $A$  e  $B$  se esiste un omeomorfismo  $\phi : A \rightarrow B$ , cioè una funzione continua con inversa continua. Intuitivamente un omeomorfismo è una deformazione continua, ad esempio allungare, torcere e flettere una figura sono omeomorfismi, mentre non lo sono strapparla o incollarne due insieme. Ma con essa si verifica il problema opposto: due oggetti molto diversi dal punto di vista intuitivo possono risultare omeomorfi. Ad esempio una tazza con manico è omeomorfa ad una ciambella.

Per questo motivo utilizzeremo l'**omologia persistente**. Una differenza sostanziale rispetto alla geometria e alla topologia è che il concetto di forma è legato non solo allo spazio  $X$  che stiamo studiando, ma anche ad una funzione  $f$  continua, definita su  $X$  e solitamente a valori reali, detta funzione filtrante. Essa trasmette l'idea di forma dell'osservatore tenendo traccia del cambiamento dell'immagine sotto tale funzione e rilevando quali caratteristiche topologiche persistono. Prendiamo nuovamente in considerazione una tazza rovesciata ed una ciambella e come funzione filtrante prendiamo l'ordinata, allora l'omologia persistente riesce a distinguere i due oggetti (figure 2.1 e 2.2). Ma se con la stessa funzione consideriamo una tazza ed una caraffa rovesciate esse risultano molto simili. Infatti con questa funzione filtrante abbiamo individuato se questi oggetti hanno un manico e se possono contenere del liquido, trascurando altre informazioni che si possono ottenere con altre funzioni [1], [2].

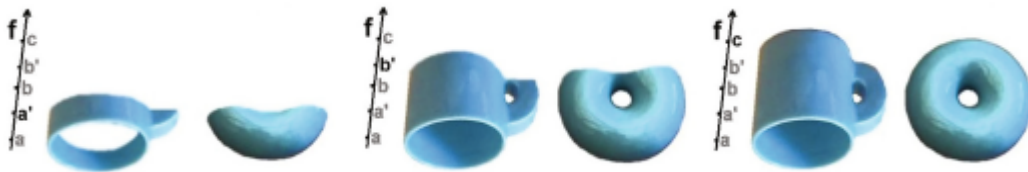


Figura 2.1: Insiemi di sottolivello di una tazza e di una ciambella.

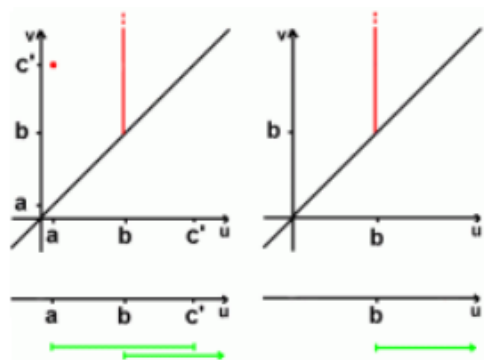


Figura 2.2: Diagramma di persistenza e barcode della tazza e della ciambella.

Ora diamo le definizioni precise.

## 2.1 Omologia simpliciale

In generale è molto difficile calcolare l'omologia di un oggetto topologico arbitrario. Dunque lo approssimeremo con un complesso simpliciale e poi calcoleremo l'omologia che in questo caso si dice omologia simpliciale. I contenuti di questo capitolo sono tratti da [8], [10] e [11].

**Definizione 2.1.** Chiamiamo *combinazione affine* di  $u_0, \dots, u_k \in \mathbb{R}^d$   $\sum_{i=0}^k \lambda_i u_i$  con  $\sum_{i=0}^k \lambda_i = 1$ .

**Definizione 2.2.** L'*inviluppo affine* di  $u_0, \dots, u_k \in \mathbb{R}^d$  è l'insieme delle combinazioni affini di  $u_0, \dots, u_k$ .

**Definizione 2.3.** I punti  $u_0, \dots, u_k \in \mathbb{R}^d$  si dicono *affinemente indipendenti* se  $\sum_{i=0}^k \lambda_i u_i = \sum_{i=0}^k \mu_i u_i$  con  $\sum_{i=0}^k \lambda_i = \sum_{i=0}^k \mu_i = 1$  implica  $\forall i \lambda_i = \mu_i$ .

**Definizione 2.4.** Una *combinazione convessa* è una combinazione affine con  $\lambda_i \geq 0 \forall i$

**Definizione 2.5.** L'*inviluppo convesso*  $\text{conv}(u_0, \dots, u_k)$  dei punti  $u_0, \dots, u_k \in \mathbb{R}^d$  è l'insieme di tutte le combinazioni convesse dei punti  $u_0, \dots, u_k$ .

**Proposizione 2.1.1.**  $\text{conv}(u_0, \dots, u_k)$  è il più piccolo insieme convesso contenente  $u_0, \dots, u_k$ .

**Definizione 2.6.** Chiamiamo *k-simplesso*  $\sigma$  l'inviluppo convesso di  $k+1$  punti affinemente indipendenti.

La *dimensione* di un  $k$ -simplesso è  $k$ .

L'*insieme vuoto* è per definizione l'unico  $(-1)$ -simplesso.

I  $k+1$  punti si dicono *vertici* di  $\sigma$ .

**Esempio 2.1.** • Un punto isolato è uno 0-simplesso.

- Un segmento è un 1-simplesso.
- Un triangolo è un 2-simplesso.
- Un tetraedro è un 3-simplesso.

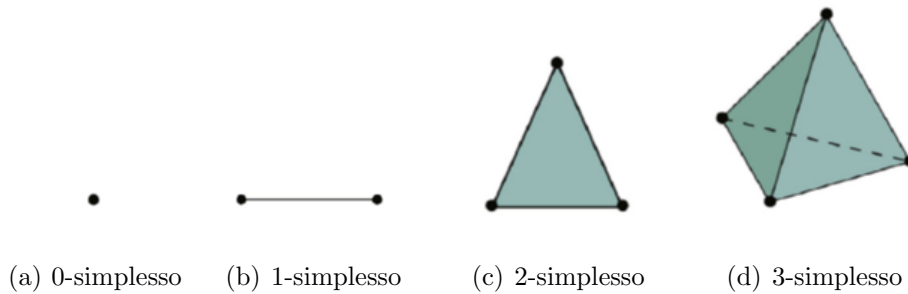


Figura 2.3: Alcuni esempi di semplici.

**Definizione 2.7.** La *faccia*  $\tau$  di un semplice  $\sigma$  generato dai punti  $u_0, \dots, u_k$  è l'involuppo convesso di un sottoinsieme di  $\{u_0, \dots, u_k\}$ .

Se  $\tau$  è una faccia di  $\sigma$  scriviamo  $\tau \subseteq \sigma$ . Se  $\tau$  è una faccia propria di  $\sigma$ , cioè  $\tau \subseteq \sigma$  e  $\tau \neq \sigma$ , scriviamo  $\tau \subset \sigma$ .

$\sigma$  è una *cofaccia* di  $\tau$ .

**Definizione 2.8.** Una *orientazione* di un  $k$ -simple  $\sigma = \{u_0, \dots, u_k\}$ , indicata con  $[\sigma]$ , è una classe di equivalenza dell'ordine dei vertici di  $\sigma$ , dove  $(u_0, \dots, u_k) \sim (u_{\tau(0)}, \dots, u_{\tau(k)})$  sono equivalenti se il segno di  $\tau$  è positivo.

**Definizione 2.9.** Un *complesso simpliciale geometrico finito* è una collezione finita  $K$  di semplici tale che:

1. se  $\sigma \in K$  e  $\tau \subseteq \sigma$ , allora  $\tau \in K$ ;
2. se  $\sigma_1, \sigma_2 \in K$  e  $\sigma_1 \cap \sigma_2 \neq \emptyset$  allora  $\sigma_1 \cap \sigma_2$  è faccia di  $\sigma_1$  e  $\sigma_2$ .

L'*ordine* o *dimensione* di  $K$  è la massima dimensione dei semplici che compaiono in  $K$ .

**Definizione 2.10.** Chiamiamo un insieme ordinato  $\{\sigma_1, \dots, \sigma_m\}$  *filtro* se è tale che se  $\sigma_i$  è una faccia propria di  $\sigma_j$  allora  $i < j$ .

**Definizione 2.11.** Se  $K_i$  sottoinsieme di un complesso simpliciale  $K$ ,  $K_i \subseteq K$ , è un simpliciale complesso si dice *sottocomplesso* di  $K$ .



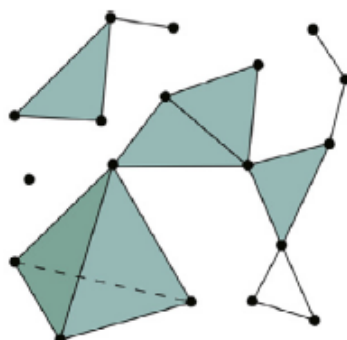


Figura 2.4: Complesso simpliciale di ordine 3.

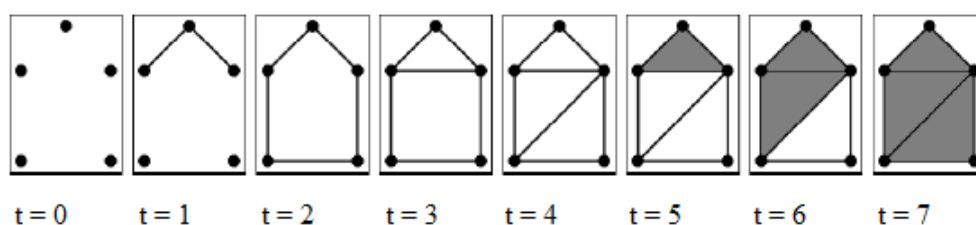


Figura 2.5: Esempio di filtrazione.

**Definizione 2.12.** Una *filtrazione* è una catena di sottocomplessi.

$$\emptyset = K_0 \subset K_1 \subset \cdots \subset K_m = K \quad (2.1)$$

**Definizione 2.13.** Sia  $K$  un complesso simpliciale. Una *funzione filtrante*  $f$  è una funzione  $f : K \rightarrow \mathbb{R}$  usata per costruire una filtrazione, tale che ogni valore reale assegnato ad un semplice di  $K$  non è più piccolo di quello assegnato alle sue facce.

Sia  $f_i = f(\sigma_i)$  e sia  $f_0 = -\infty$ , allora definiamo  $K_i = f^{-1}(-\infty, f_i]$ .

Nel corso dell'evoluzione della filtrazione alcune caratteristiche importanti dell'oggetto di studio, come la connessione per archi e la presenza di buchi che sono rilevate dai numeri di Betti che vedremo nel prossimo paragrafo, possono comparire e scomparire. La persistenza di alcune di esse determina la costruzione della struttura finale dell'immagine.

## 2.2 Omologia persistente

Come abbiamo detto le principali connotazioni che consideriamo di una immagine sono misurate dai numeri di Betti. Per poterli definire abbiamo prima bisogno di sapere cos'è un gruppo di omologia simpliciale.

**Definizione 2.14.** Dato un complesso simpliciale  $K$  definiamo il *gruppo della  $d$ -catena*  $(C_d(K), +)$  come il gruppo abeliano libero in cui gli elementi sono i  $d$ -simplessi orientati di  $K$ , dove  $[\sigma] = -[\tau]$  se  $\sigma = \tau$  e  $\sigma$  e  $\tau$  sono orientate in senso opposto.

**Definizione 2.15.** Un *gruppo abeliano libero* è un gruppo abeliano con una base.

Si dice *base* di un gruppo un suo sottoinsieme tale che ogni elemento del gruppo è univocamente determinato da una combinazione lineare degli elementi della base.

In questo caso la base è formata dai  $d$ -simplessi orientati modulo l'identificazione di un semplice orientato  $s$  con  $-s'$  se  $s'$  è il medesimo semplice con l'orientazione opposta.

**Definizione 2.16.** Un elemento di  $C_d(K)$  è una  *$d$ -catena* di simplessi, cioè una combinazione lineare formale  $\sum_{i=1}^k a_i[\sigma_i]$ , dove  $a_i \in \mathbb{Z}$  e  $\sigma_i$  è un  $d$ -simpleso del complesso simpliciale geometrico considerato.

**Definizione 2.17.** Il  *$d$ -esimo operatore di bordo* è un omeomorfismo  $\partial_d : C_d(K) \rightarrow C_{d-1}(K)$  tale che  $\partial_d[u_0, \dots, u_d] = \sum_{i=0}^d (-1)^i [u_0, \dots, \hat{u}_i, \dots, u_d]$ , dove  $\hat{u}_i$  denota la cancellazione del vertice  $u_i$  dalla catena. Per linearità tale definizione si estende in modo unico a  $C_d(K)$ . Otteniamo in questo modo una catena di omeomorfismi:

$$\dots \rightarrow C_{d+1}(K) \xrightarrow{\partial_{d+1}} C_d(K) \xrightarrow{\partial_d} C_{d-1}(K) \rightarrow \dots$$

Definiamo

1.  $Z_d(K) := \ker \partial_d$ , ogni suo elemento si chiama  *$d$ -ciclo*;

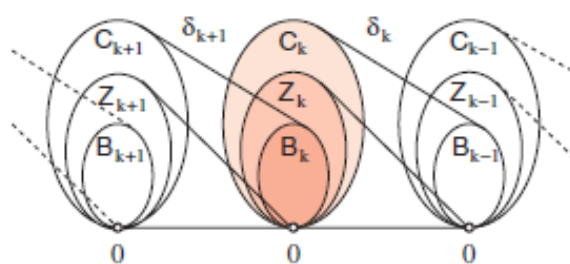
2.  $B_d(K) := \text{Im} \partial_{d+1}$ , ogni suo elemento si chiama  $d$ -bordo.

Per la linearità di  $\partial_d$  si ha che  $(Z_d(K), +)$  e  $(B_d(K), +)$  sono gruppi abeliani.

**Lemma 2.2.1.** *Una importante proprietà dell'operatore di bordo è:*

$$\partial_d \partial_{d+1} = 0.$$

Una conseguenza di questo fatto è che  $B_d(K) \subseteq Z_d(K) \subseteq C_d(K)$  come sottogruppi.



**Definizione 2.18.** Il  $d$ -esimo gruppo di omologia simpliciale è

$$H_d(K) := Z_d(K) / B_d(K).$$

Le classi di omologia sono rappresentate dai cicli che non sono bordi. Due cicli sono omologhi se la loro differenza è un bordo. Consideriamo la figura

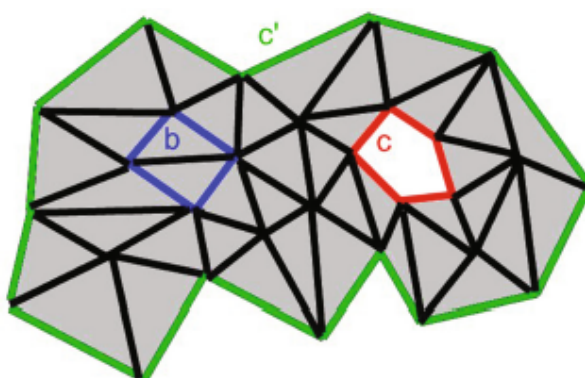


Figura 2.6: Complesso simpliciale formato dai triangoli grigi e le loro facce.

2.6 le catene blu  $b$ , rossa  $c$  e verde  $c'$  sono dei cicli. La catena  $b$  è anche un bordo. Invece le catene  $c$  e  $c'$  prese singolarmente non lo sono, poiché il bordo della figura è la catena rossa  $c$  insieme a quella verde  $c'$ . Dunque le due catene  $c$  e  $c'$  sono omologhe.

**Definizione 2.19.** Il rango del gruppo di omologia  $\beta_d(K) := \text{rg}H_d(K)$  viene chiamato *d-esimo numero di Betti*.

Vediamo alcuni esempi del numero di Betti:

- Esempio 2.2.**
1.  $\beta_0(K)$  conta le componenti connesse di  $K$ ;
  2.  $\beta_1(K)$  conta i buchi di  $K$ ;
  3.  $\beta_2(K)$  conta gli spazi vuoti di  $K$ .

## 2.3 Barcode e diagramma di persistenza

Consideriamo la catena di inclusioni 2.1, per ogni  $i, j$  tali che  $0 \leq i < j \leq m$  si ha che l'inclusione

$$I^{i,j} : K_i \hookrightarrow K_j$$

è una applicazione simpliciale e induce un omomorfismo di gruppi

$$F_d^{i,j} : H_d(K_i) \rightarrow H_d(K_j)$$

per ogni  $0 \leq i < j \leq m$ , dove  $H_d(K_i)$  e  $H_d(K_j)$  sono  $d$ -esimi gruppi di omologia simpliciale.

Il numero di Betti  $\beta_d^{i,j}$  è il rango di  $\text{Im}(F_d^{i,j})$ .

**Definizione 2.20.** Diciamo che una classe  $h$  "nasce" al tempo  $f_i$  se  $h \in H_d(K_i)$  e  $h \notin \text{Im}(F_d^{i-1,j})$ , cioè se è una caratteristica nuova.

Diciamo che "muore" al tempo  $f_j$  se  $F_d^{i,j-1}(h) \notin \text{Im}(F_d^{i-1,j-1})$  ma  $F_d^{i,j}(h) \in \text{Im}(F_d^{i-1,j})$ , cioè se al tempo  $f_j$  si unisce ad un'altra caratteristica.

Il tempo tra la nascita e la morte di una caratteristica viene chiamato *tempo di persistenza*.

Da queste informazioni possiamo definire un *barcode* in cui sono rappresentate tramite dei vettori la nascita e la morte delle diverse caratteristiche. Innanzitutto vediamo come si costruiscono i barcode della filtrazione 2.5. Al tempo  $t = 0$  ci sono 5 punti disconnessi quindi nel barcode di  $\beta_0$  2.7 nascono 5 linee in  $t = 0$ . Al passo successivo si aggiungono due lati e le componenti connesse per archi diminuiscono a 3, dunque nel grafico vedremo 2 linee fermarsi. Dal tempo  $t = 2$  vi è un'unica componente connessa per archi che rimane per sempre, questo lo indichiamo con una freccia a destra di quella barra.

Per  $\beta_1$  ragioniamo allo stesso modo prendendo in considerazione i buchi 1-dimensionali. Il primo buco compare al tempo  $t = 2$ , infatti i 5 lati che si sono aggiunti in questi passaggi formano una linea spezzata chiusa. Quindi nel barcode la prima barra nasce in  $t = 2$ . Nel passo seguente l'aggiunta di un lato divide il primo buco in due e dunque nasce un'altra barra nel barcode. Analogamente per  $t = 4$ . Quando  $t = 5$  il triangolo superiore si riempie e ciò provoca la morte di un buco e allo stesso modo in  $t = 6$  e  $t = 7$ .

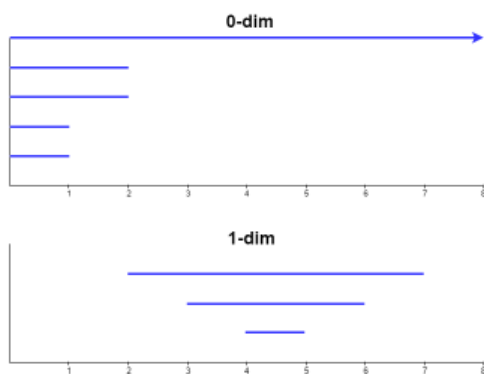


Figura 2.7: Barcode della filtrazione 2.5: in alto vediamo il barcode di  $\beta_0$ , mentre in basso quello di  $\beta_1$ .

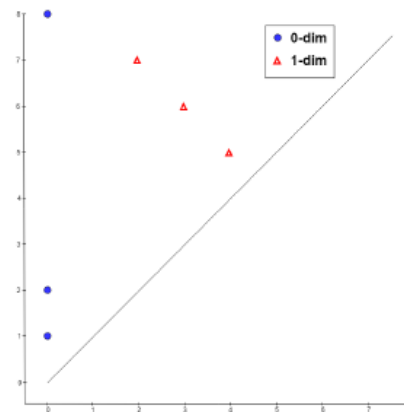


Figura 2.8: Diagramma di persistenza della filtrazione 2.5.

Un diagramma di persistenza è simile ad un barcode. Ogni linea di un barcode è univocamente determinata da i suoi punti di inizio e di fine, rappresentiamo in questo modo una barra tramite il punto (inizio, fine) in un

piano Euclideo e il grafico così ottenuto si chiama *diagramma di persistenza*. Il diagramma di persistenza della filtrazione 2.5 dunque è dato da 2.8.

## 2.4 Entropia persistente

Una volta ottenuti i barcode è possibile calcolare l'entropia persistente [14].

**Definizione 2.21.** Dato un complesso simpliciale filtrato  $\{K(t) : t \in F\}$  e il corrispondente barcode  $B = \{a_i = [x_i, y_i) : i \in I\}$ , dove  $F$  e  $I$  sono opportuni insiemi di indici, l'*entropia persistente*  $E$  del complesso simpliciale filtrato è calcolata nel modo seguente:

$$E = - \sum_{i \in I} p_i \log(p_i)$$

dove  $p_i = \frac{l_i}{L}$ ,  $l_i = y_i - x_i$  e  $L = \sum_{i \in I} l_i$ .

Se un intervallo non ha il punto finale,  $[x_i, +\infty)$ , nel barcode verrà considerato  $[x_i, m)$ , dove  $m = \max\{F\} + 1$ .

La stabilità è garantita dal seguente teorema.

**Teorema 2.4.1.** *Siano date due funzioni,  $f : V \rightarrow \mathbb{R}$  e  $g : V \rightarrow \mathbb{R}$ , definite su un insieme finito di vertici  $V$  di  $\mathbb{R}^n$ , allora per ogni  $\varepsilon > 0$ , esiste  $\delta > 0$  tale che*

$$\|f - g\|_\infty \leq \delta \Rightarrow |E(f) - E(g)| \leq \varepsilon.$$

# Capitolo 3

## Applicazioni

I contenuti di questo capitolo sono tratti dall'articolo [8].

Il metodo corrente usato per analizzare dati consiste di tre passaggi.

1. Si strutturano i dati mediante un complesso simpliciale.
2. Si esaminano mediante una filtrazione predefinita. Viene utilizzata una modalità con più filtrazioni per incrociare diversi tipi di informazioni.
3. Gli intervalli di persistenza sono ottenuti da omologia persistente, che può essere visualizzata con un barcode. La somiglianza della forma viene misurata in base alla distanza tra i barcode.

Vediamo in dettaglio l'implementazione. Consideriamo l'oggetto in input come insieme di punti di cui calcoliamo una triangolazione e costruiamo un complesso di Vietoris-Rips.

### 3.1 Triangolazione di Delaunay

**Definizione 3.1.** Una *triangolazione* di un insieme finito di punti  $P \subset \mathbb{R}^2$  è una collezione  $\mathcal{T}$  di triangoli tali che:

- $\text{conv}(P) = \bigcup_{T \in \mathcal{T}} T$ .
- I vertici di tutti i triangoli di  $\mathcal{T}$  sono punti in  $P$ .

- Per ogni  $p \in P$ ,  $t \in \mathcal{T}$ , se  $p \in t$  allora  $p$  è un vertice di  $t$ .
- Per ogni coppia di triangoli  $T, U \in \mathcal{T}$  l'intersezione  $T \cap U$  è un vertice in comune, un lato in comune o vuota.

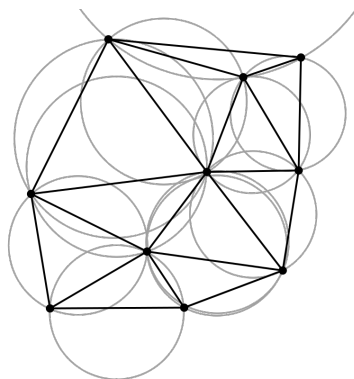
Dato un generico insieme di punti esiste almeno una triangolazione.

Una importante proprietà è la seguente:

**Proposizione 3.1.1.** *Ogni insieme  $P \subset \mathbb{R}^2$  di  $n \geq 3$  punti ammette una triangolazione a meno che tutti i punti in  $P$  siano collineari.*

Esaminiamo un particolare tipo di triangolazione molto utilizzata nel trattamento di immagini.

**Definizione 3.2.** Una triangolazione di un insieme finito di punti  $P \subset \mathbb{R}^2$  viene detta di *Delaunay* se il cerchio circoscritto ad ogni triangolo è vuoto, ovvero nessun punto di  $P$  vi giace all'interno.



Esistono tantissime triangolazioni diverse, ma la triangolazione di Delaunay gode di proprietà per le quali è quella che più ragionevolmente si avvicina all'oggetto che si vuole approssimare. Essa massimizza l'ampiezza del più piccolo angolo interno tra tutte le triangolazioni possibili. Questo non implica che non esistano triangoli "allungati", ma in questo caso questi saranno presenti anche in tutte le altre triangolazioni.



**Esempio 3.1.** Si verifica un caso particolare se prendiamo in considerazione 4 punti non allineati. Ci sono due possibili triangolazioni: in generale una sola è una triangolazione di Delaunay 3.1(a), se però i 4 punti giacciono sulla stessa circonferenza (nel qual caso i punti si dicono concentrici) allora entrambe sono triangolazioni di Delaunay valide 3.1(b).

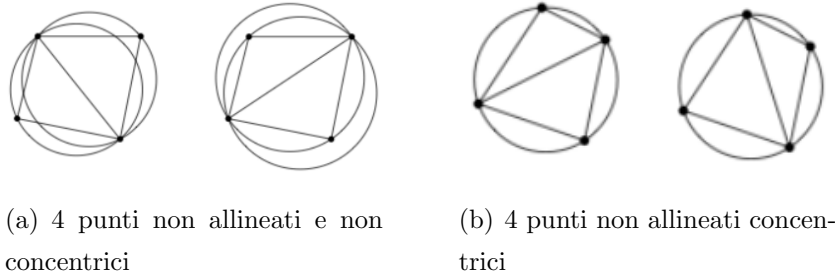


Figura 3.1: Possibili triangolazioni per 4 punti non allineati. In 3.1(a) solo la triangolazione a sinistra è di Delaunay, mentre in 3.1(b) sono entrambe triangolazioni di Delaunay.

Valgono i seguenti risultati:

**Proposizione 3.1.2.** *Dato un insieme  $P \subset \mathbb{R}^2$  di quattro punti non tutti collineari e non concentrici, allora  $P$  ha esattamente una triangolazione di Delaunay.*

**Teorema 3.1.3.** *Sia  $P \subset \mathbb{R}^2$  un insieme di  $n$  punti non tutti collineari, allora esiste una triangolazione di Delaunay.*

**Definizione 3.3.** Sia  $S$  un insieme di punti in  $\mathbb{R}^2$  o  $\mathbb{R}^3$  in posizione generale e sia  $\Omega = Conv(S)$ . Sia  $\Omega$  un dominio chiuso e limitato di  $\mathbb{R}^2$  o di  $\mathbb{R}^3$ . Una *mesh* di  $\Omega$  è una triangolazione di  $\Omega$ , più precisamente  $T$  si dice *mesh* di  $\Omega$  se:

- $\Omega = \bigcup_{K \in T} K$ .
- Ogni elemento di  $K$  in  $T$  è non vuoto.
- L'intersezione dell'interno di ogni coppia di elementi è un insieme vuoto.

- L'intersezione di due elementi di  $T$  è un vertice, un lato, una faccia oppure vuota.

La triangolazione 3D di Delaunay è l'input dell'algoritmo utilizzato in [12] per riconoscere le espressioni. I vertici corrispondono a punti di riferimento antropometrici e i lati alle loro distanze.

## 3.2 Complesso di Vietoris-Rips

Vediamo ora un metodo per approssimare un oggetto tridimensionale qualsiasi con un complesso simpliciale al fine di capirne la topologia. Campioniamo un insieme di punti e costruiamo un complesso simpliciale i cui vertici sono i punti campionati e, fissato  $\varepsilon > 0$ , uniamo due vertici con un lato se la loro distanza è minore di  $\varepsilon$ . Quando tre punti sono collegati a due a due anche il triangolo corrispondente fa parte del complesso simpliciale. Analogamente un simpleso di ordine superiore fa parte del complesso simpliciale se ogni suo vertice è distante dagli altri vertici del simpleso meno di  $\varepsilon$ . Il complesso così costruito si chiama complesso di Vietoris-Rips [13]. Consideriamo innanzitutto  $\varepsilon = 0$  e in seguito prendiamo valori sempre più grandi. Abbiamo costruito una filtrazione grazie alla funzione distanza.

**Esempio 3.2.** Vogliamo studiare la topologia di una corona circolare che ha il raggio della circonferenza  $r = 1$  e il raggio della circonferenza esterna  $R = 2$  [13].

Nella figura 3.2(a) sono rappresentati i punti campione della corona circolare e il complesso simpliciale è l'unione di essi. Nelle figure seguenti  $\varepsilon$  aumenta, quindi alcuni punti solo collegati da dei lati. Se costruiamo il barcode per  $\beta_0$  inizialmente avremo numerose barre, infatti i punti sono sconnessi, ma quando  $\varepsilon$  prende valori più grandi le barre si fermano e per  $\varepsilon = 0.8$  abbiamo una sola barra ancora in vita, che rimane anche per valori maggiori 3.3(a). In 3.2(c) nasce un percorso chiuso tra alcuni vertici evidenziato in rosso, che determina la nascita della prima barra nel barcode di  $\beta_1$ . Come vediamo in 3.3(b) altri cicli nascono e muoiono ma il più significativo è quello che nasce

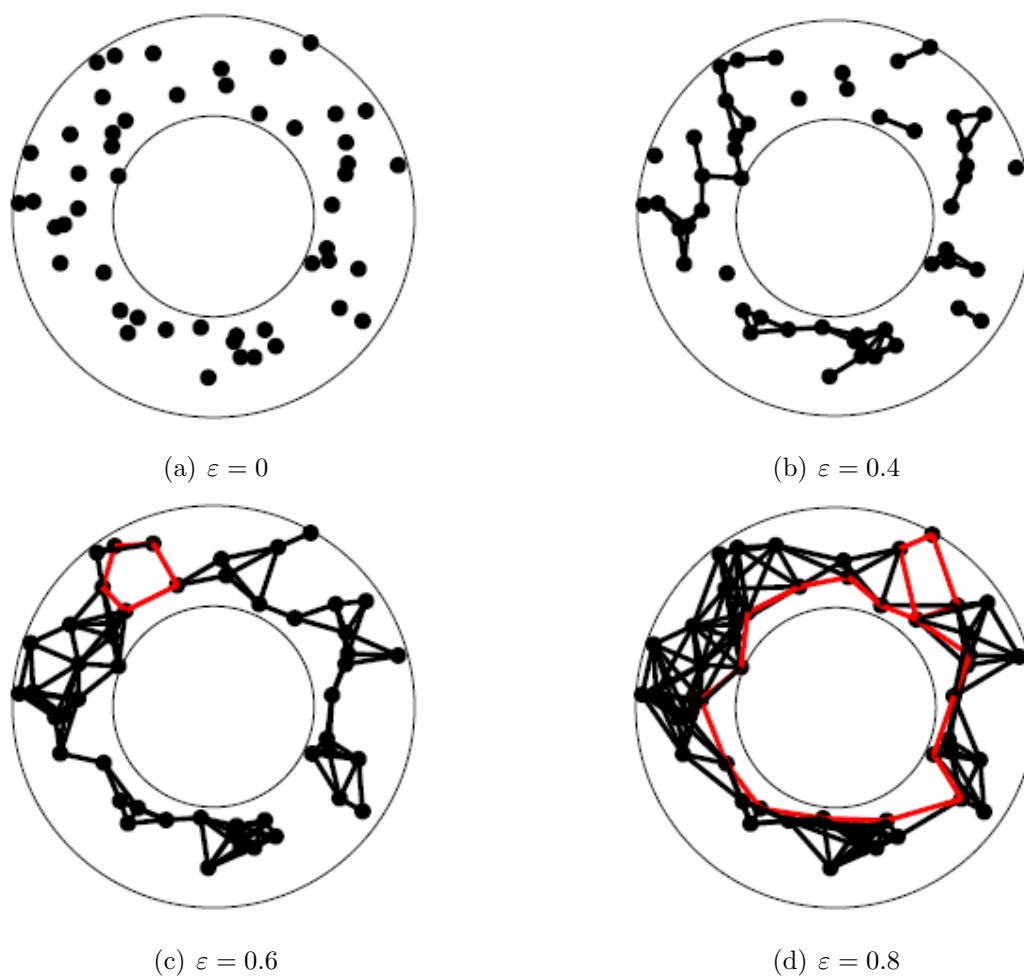


Figura 3.2: Approssimazione di una corona circolare con il metodo di Vietoris-Rips.

al tempo 0.74 e muore al tempo 2.11 che rappresenta il buco interno alla corona. Da questi diagrammi possiamo dedurre che la figura è connessa ed ha un buco, che è proprio la topologia della corona circolare. Infatti in un barcode certe barre possono essere generate dal rumore di campionamento. Generalmente in questi casi la nascita e la morte avvengono in due tempi molto ravvicinati, ciò significa che le barre sono molto brevi. In un diagramma di persistenza i punti generati dal rumore si trovano molto vicini alla diagonale. In questo esempio vediamo nella figura 3.3 che c'è una barra molto lunga per

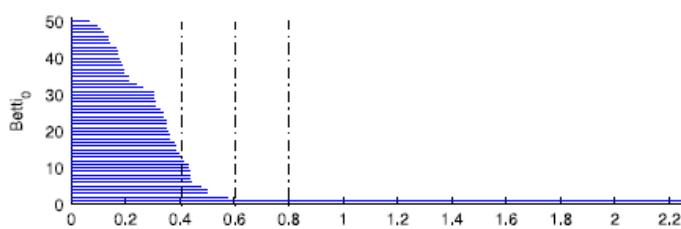
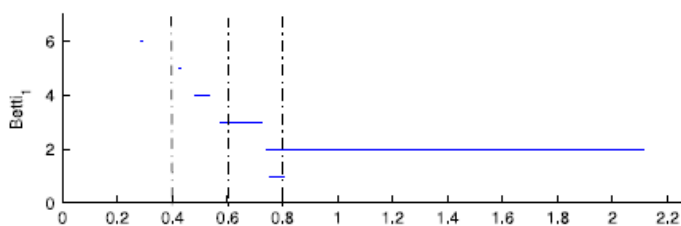
(a) Barcode  $\beta_0$ (b) Barcode  $\beta_1$ 

Figura 3.3: Barcode della corona circolare della figura 3.2.

$\beta_0$ , quindi la figura è connessa; per  $\beta_1$  la barra più importante indica il buco presente nella corona, mentre le altre sono generate dal rumore.

### 3.3 Funzioni filtranti

Per il secondo punto sorge spontaneo chiedersi con quale funzione lavorare. Analizziamo la figura 3.4, in cui un complesso simpliciale è stato filtrato con tre diverse funzioni. È molto utile ricorrere a più filtri per ottenere maggiori informazioni, poiché ogni barcode ha una certa importanza e fornisce determinate informazioni.

Sia  $X = \{x_j\}_{j=1,\dots,m}$  un insieme di  $m$  punti. Consideriamo una famiglia di  $n$  funzioni  $\{f_i : X \rightarrow \mathbb{R}\}_{i=1,\dots,n}$ , che rappresentiamo in una matrice in cui ogni riga è relativa ad una diversa funzione filtrante ed ogni colonna ad un punto dell'insieme  $X$ .

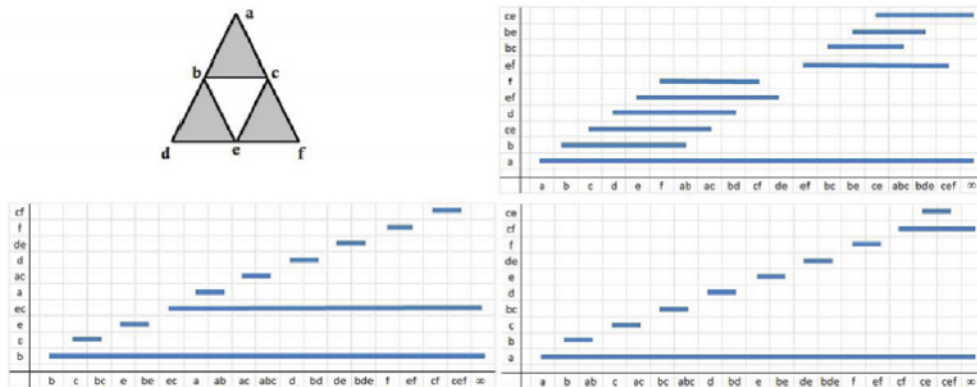


Figura 3.4: In alto a sinistra troviamo il complesso simpliciale di cui sono mostrati tre barcode. Nell'asse orizzontale sono segnati i simpliciali ordinati e nell'asse verticale sono rappresentati gli intervalli.

$$\begin{pmatrix} f_{1,1} & f_{1,2} & \cdots & f_{1,m} \\ f_{2,1} & f_{2,2} & \cdots & f_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n,1} & f_{n,2} & \cdots & f_{n,m} \end{pmatrix}$$

Ogni riga della matrice è una distribuzione di  $X$  sotto la funzione corrispondente. Per ogni funzione consideriamo la filtrazione data dall'evoluzione di  $f^{-1}([\alpha, +\infty))$ , dove  $\alpha$  assume valori sempre più piccoli. Costruiamo i barcode in diverse dimensioni  $\{B_{i,j}\}_{j=1,\dots,d}$ , dove  $d$  è la più alta dimensione del barcode. Possiamo rappresentare in una matrice i barcode ottenuti:

$$\begin{pmatrix} B_{1,0} & B_{1,1} & \cdots & B_{1,d} \\ B_{2,0} & B_{2,1} & \cdots & B_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ B_{n,0} & B_{n,1} & \cdots & B_{n,d} \end{pmatrix}$$

Per misurare la distanza tra barcode possiamo usare un metodo che li quantizza. Nella figura 3.5 vediamo due barcode, sopra vi è un barcode di dimensione 0 e sotto di dimensione 1, che sono stato divisi in 7 bin, come mostrato dalle linee tratteggiate. Nella tabella a destra viene registrato il

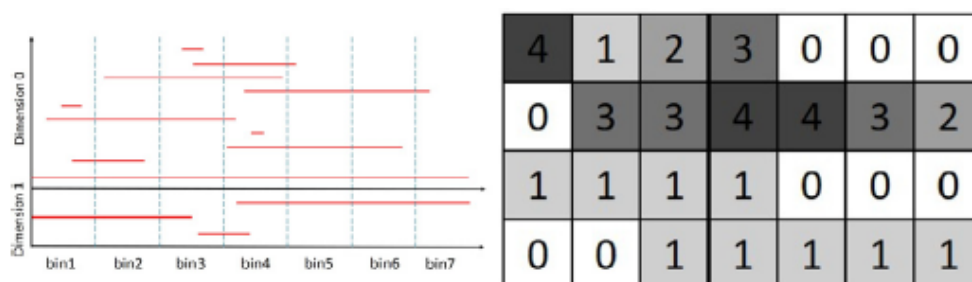


Figura 3.5: A sinistra in alto è presente un barcode di dimensione 0 e in basso un barcode di dimensione 1, entrambi divisi in 7 bin. A destra vi è una tabella  $4 \times 7$  in cui le colonne indicano i 7 bin e nelle righe è registrato il numero degli intervalli che cominciano, continuano o terminano in ciascun bin. Più precisamente le prime due righe si riferiscono al barcode 0-dimensionale e le altre due a quello 1-dimensionale. Nella riga superiore, in entrambi i casi, sono indicati gli intervalli che cominciano, mentre in quella inferiore quelli che continuano o terminano.

numero degli intervalli che nascono, passano o muoiono in una zona. Negli esperimenti i confronti vengono fatti utilizzando le tabelle così ottenute.

### 3.4 Esperimenti

Il metodo descritto è stato applicato al riconoscimento di espressioni facciali [8]. È stato utilizzato il database di espressioni facciali 3D FaceWarehouse [9], che contiene 150 individui e per ognuno sono presenti 20 espressioni distinte 3.6. Vengono adottate delle procedure standard per diminuire la sensibilità alla deformazione spaziale. Nella figura 3.7 è mostrato il processo. Si considera solo la parte anteriore della testa dato che la parte posteriore non è rilevante per l'analisi delle espressioni. Successivamente il volto viene diviso in tre parti lungo l'asse orizzontale ed infine anche lungo l'asse verticale. Il viso, le tre parti orizzontali e le tre parti verticali formano l'input dell'algoritmo utilizzato in questo esperimento. Per ogni espressione sono state adoperate delle funzioni distanza da piani fissati, nello specifico  $\rightarrow$  indica la

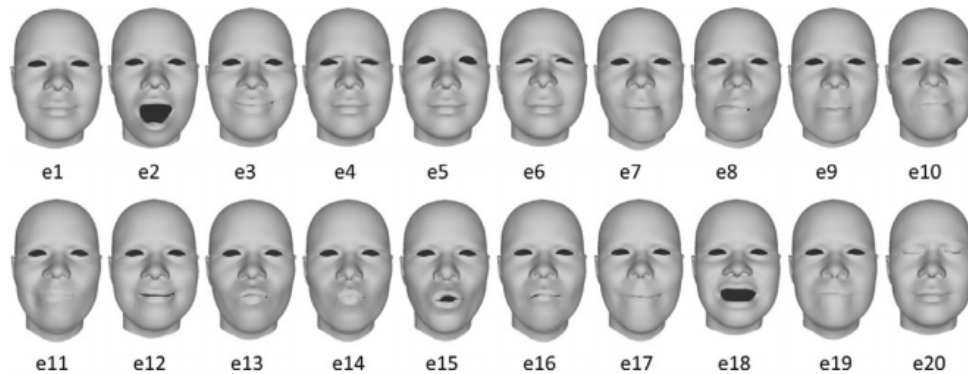


Figura 3.6: Esempio di 20 espressioni presenti nel database FaceWarehouse.

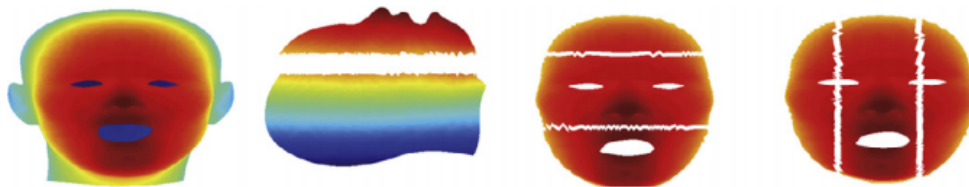


Figura 3.7: La prima immagine mostra il modello della testa 3D. Le successive tre immagini mostrano la sua suddivisione nelle parti che formano l'input dell'algoritmo.

distanza di ciascun punto dell'immagine tridimensionale da un piano verticale situato a sinistra della figura, similmente  $\leftarrow$ ,  $\uparrow$ ,  $\downarrow$ ,  $\nearrow$ ,  $\swarrow$ ,  $\nwarrow$ ,  $\searrow$ ,  $\odot$  indica la distanza dal baricentro e  $\otimes$  la distanza dal bordo. Sono ben rappresentate nella figura 3.8. Non otteniamo alcun guadagno con le funzioni  $\cdot$  e  $*$ , che indicano la distanza dei punti da un piano parallelo al foglio che si trova rispettivamente dietro e davanti alla figura, per cui non sono state usate. Per ciascuna funzione è stata costruita una filtrazione. In questo esperimento viene utilizzata la Support Vector Machine (SVM), un metodo che da pochi decenni si dimostra molto efficace nelle classificazioni anche quando non c'è un iperpiano separatorio. 80% dei volti di ogni classe viene utilizzato per il training set, mentre il restante 20% serve per i test. Fissata una determinata funzione filtrante calcoliamo il barcode di una certa espressione e misuriamo la distanza tra i barcode di questa immagine e quelli ottenuti da un training

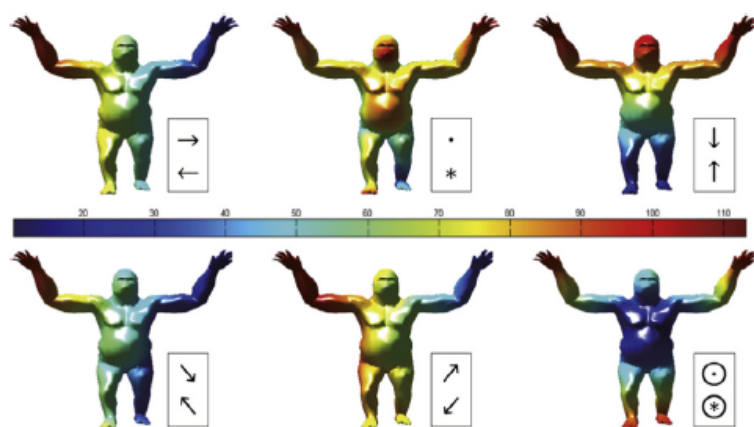


Figura 3.8: Sono rappresentate le funzioni distanza.

set. Per ogni funzione, otteniamo una rappresentazione a matrice. Vettorizziamo questa matrice e la inseriamo come colonne di una nuova matrice, che tiene conto di tutte le funzioni.

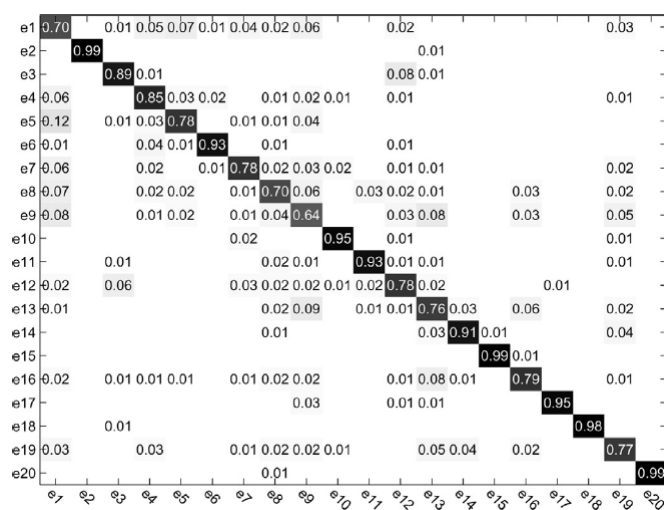


Figura 3.9: Matrice di confusione del riconoscimento di espressioni facciali 3D del database FaceWarehouse.

È stato eseguito diverse volte e i risultati così ottenuti hanno una precisione media di 85.8%. Nella matrice di confusione 3.9 vediamo risultati più dettagliati. Osserviamo che alcune espressioni hanno una precisione molto



alta (e2, e15, e18, e20 hanno una accuratezza superiore al 95%). Queste espressioni infatti si differenziano molto dalle altre per l'apertura della bocca o per gli occhi chiusi. Dunque i diagrammi di persistenza saranno molto riconoscibili. L'accuratezza più bassa si riscontra in e9 con 64%, infatti questa espressione schiude leggermente le labbra e si confonde con l'espressione neutrale e1 e con e13.

Il testo [12] descrive una piattaforma multisensoriale, chiamata *Wize Mirror*, che contiene video, immagini e scansioni 3D del volto al fine di monitorare i segni correlati al rischio cardiometabolico. L'algoritmo utilizzato prende in input la triangolazione di Delaunay. Vediamo come esempio il riconoscimento degli sbadigli. *Wize Mirror* li identifica abbinando le caratteristiche geometriche basate su punti di riferimento di ciascun fotogramma in una sequenza video con modelli che rappresentano sbadigli ed espressioni neutre. Nella figura 3.10 è mostrata la probabilità di ogni fotogramma di una sequenza video di rappresentare uno sbadiglio. Notiamo che i massimi sono posizionati correttamente in corrispondenza degli sbadigli.

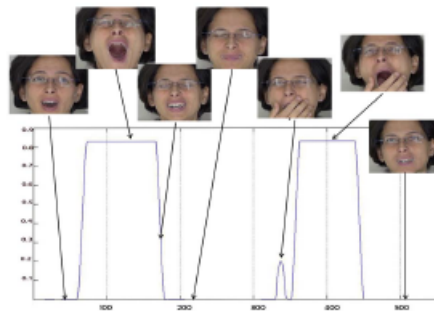


Figura 3.10: Probabilità di uno sbadiglio in una sequenza video.

### 3.5 Altre ricerche

Altre ricerche trattano il riconoscimento e la classificazione delle emozioni attraverso esperienze sensoriali uditive [14]. Vengono prese in considerazio-

ne alcune caratteristiche fisiche fondamentali del suono, come la frequenza, l'intensità e la durata, che fanno parte del concetto di prosodia; queste attribuiscono ad una frase, a seconda del modello prosodico, diverse intenzioni emotive.

Il segnale acustico viene campionato e, presa una filtrazione adeguata, si calcola il relativo barcode. Quindi viene calcolata l'entropia persistente e si classificano i segnali usando SVM, in particolare vengono testate diverse macchine vettoriali di supporto con kernel diversi e viene scelto quello con una maggiore precisione.

In questo caso la precisione viene considerata come la percentuale di dati ben classificati in un insieme di dati:

$$accuracy = \frac{m}{n},$$

dove  $m$  è il numero di dati ben classificati e  $n$  è la dimensione dell'intero insieme di dati utilizzato nel test.

Il metodo è stato applicato ad un insieme di dati RAVDESS [15] composto da 1440 audio: 24 attori interpretano 60 audio di emozioni (emozione neutra, calma, felicità, tristezza, rabbia, paura, disgusto e sorpresa). Sono stati eseguiti tre esperimenti con questo insieme di dati.

Nel primo ogni valore di entropia persistente è un punto del training dataset. La precisione ottenuta con un kernel lineare è del 20,3%. Osserviamo da questo fallimento che il problema del riconoscimento delle emozioni è multi-dimensionale.

Nel secondo esperimento ogni punto del dataset è un vettore di 24 elementi che corrispondono ai valori di entropia persistente della stessa emozione interpretata da 24 attori diversi. Il dataset è stato diviso: 40 punti sono stati utilizzati per il training set, mentre 20 per i test. In questo caso è stato usato un kernel gaussiano. La precisione raggiunta è di 92,5% nel training set e 90% nell'insieme di dati per il test; per l'insieme completo la precisione è di 96,66%. Però presenta alcuni inconvenienti: ogni emozione ha bisogno di 24 elementi e per il modo in cui sono state raggruppati i dati è rimasto un insieme piccolo per il test.

Nel terzo esperimento ogni punto del dataset è costituito da un vettore di 8 elementi che corrispondono al valore di entropia persistente della stessa emozione interpretata dallo stesso attore. È stato usato un kernel polinomiale di secondo grado. In 3.11 sono mostrati i valori di precisione ottenuti. Sono risultati promettenti considerando quelli presenti in letteratura precedentemente all'articolo in esame.

Feelings	Calm	Happy	Sad	Angry	Fearful	Disgust	Surprised
Calm		77.1%	68.8%	81.2%	79.2%	72.9%	60.4%
Happy			62.5%	64.6%	60.4%	58.3%	64.6%
Sad				75%	62.5%	70.8%	60.4%
Angry					68.8%	77.1%	70.8%
Fearful						72.9%	72.9%
Disgust							75%
Surprised							

Figura 3.11: Precisione della classificazione di coppie di emozioni.



# Conclusioni

In questa tesi abbiamo mostrato l'importanza del riconoscimento delle espressioni facciali ed in particolare il ruolo dell'omologia persistente. Siamo partiti dalle definizioni fondamentali dell'omologia simpliciale, che ci hanno permesso di definire strumenti molto potenti per il riconoscimento di forme: il barcode e il diagramma di persistenza. Infine abbiamo visto il metodo corrente utilizzato nelle applicazioni e alcuni esperimenti con risultati molto promettenti.



# Bibliografia

- [1] Massimo Ferri, *Persistent topology for natural data analysis - a survey*, Towards Integrative Machine Learning and Knowledge Extraction: 117-133, Springer, Cham, 2017.
- [2] Massimo Ferri, *L'incredibile ubiquità della topologia persistente*, <http://maddmaths.simai.eu/divulgazione/focus/lincredibile-ubiquita-della-topologia-persistente/>, 2016.
- [3] Rosalind W. Picard, *Affective Computing*, 1997.
- [4] Michael Feffer, Rosalind W. Picard, *A mixture of personalized experts for human affect estimation*, International Conference on Machine Learning and Data Mining in Pattern Recognition, Springer, Cham, 2018.
- [5] Ognjen Rudovic et al. *Culturenet: A deep learning approach for engagement intensity estimation from face images of children with autism*, 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2018.
- [6] Daniel Lopez Martinez, Ognjen Rudovic, Rosalind W. Picard, *Personalized automatic estimation of self-reported pain intensity from facial expressions*, 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2017.
- [7] David Su, Rosalind W. Picard, Yan Liu, *AMAI: adaptive music for affect improvement*, Ann Arbor, MI: Michigan Publishing, University of Michigan Library, 2018.

- 
- [8] Zhen Zhou, Yongzhen Huang, Liang Wang, Tieniu Tan, *Exploring generalized shape analysis by topological representations*, Pattern Recognition Letters 87: 177-185, 2017.
- [9] Chen Cao, et al. *Facewarehouse: A 3d facial expression database for visual computing*, IEEE Transactions on Visualization and Computer Graphics 20.3, 413-425, 2013.
- [10] Afra Zomorodian, Gunnar Carlsson, *Computing persistent homology*, Discrete & Computational Geometry 33.2: 249-274, 2005.
- [11] Mehmet E. Aktas, Esra Akbas, Ahmed El Fatmaoui, *Persistence Homology of Networks: Methods and Applications*, arXiv preprint arXiv:1907.08708, 2019.
- [12] Yasmina Andreu-Cabedo, et al. *Mirror mirror on the wall... An intelligent multisensory mirror for well-being self-assessment*, 2015 IEEE International Conference on Multimedia and Expo (ICME), IEEE, 2015.
- [13] Jennifer Gamble, Giseon Heo, *Exploring uses of persistent homology for statistical analysis of landmark-based shape data*, Journal of Multivariate Analysis 101.9: 2184-2199, 2010.
- [14] Rocio Gonzalez-Diaz, Eduardo Paluzo-Hidalgo, José F. Quesada, *Towards emotion recognition: A persistent entropy application*, International Workshop on Computational Topology in Image Context, Springer, Cham, 2019.
- [15] Steven R. Livingstone, Frank A. Russo, *The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multi-modal set of facial and vocal expressions in North American English*, PloS one 13.5, 2018.