

REPORT SERIES

A. Raffo, S. Biasotti

Calcolo di descrittori ibridi geometria-colore per l'analisi di similarità di forme 3D



IMATI REPORT Series

Nr. 16-05 – June 2016

Managing Editor

Paola Pietra

Editorial Office

Istituto di Matematica Applicata e Tecnologie Informatiche “E. Magenes”

Consiglio Nazionale delle Ricerche

Via Ferrata, 5/a

27100 PAVIA (Italy)

Email: reports@imati.cnr.it

<http://www.imati.cnr.it>

Follow this and additional works at: <http://www.imati.cnr.it/reports>

Copyright © CNR-IMATI, 2016.

IMATI-CNR publishes this report under the Creative Commons Attributions 4.0 license.

Calcolo di descrittori ibridi geometria-colore per l'analisi di similarità di forme 3D

Andrea Raffo^a, Silvia Biasotti^b

^a *Dipartimento di Matematica, Università degli Studi di Genova*

^b *Istituto di Matematica Applicata e Tecnologie Informatiche "E. Magenes", CNR, Genova*

Abstract.

Il riconoscimento e la classificazione di forme geometriche hanno molteplici applicazioni in grafica computazionale, spaziando dallo sviluppo di search engine all'identificazione di anomalie in modelli medicali orientati al paziente, al design di nuovi farmaci e alla sperimentazione in silico. In tutti questi contesti, il confronto tra forme viene tradotto nel confronto tra descrizioni dell'oggetto. È quindi di fondamentale importanza che una buona descrizione dell'oggetto sia quanto più possibile completa, per risultare univoca, e allo stesso tempo concisa, per poter essere facilmente computabile e confrontabile con altre descrizioni. Inoltre, la descrizione geometrica di un oggetto 3D può essere corredata da informazioni colorimetriche, eventuali annotazioni e/o dati derivanti dalla sua acquisizione/provenienza. Nell'ambito generale del confronto di forme 3D, in questo lavoro si analizza un metodo che utilizza proprietà geometriche e colorimetriche. Rispetto alla implementazione presente nella letteratura, si studia una ottimizzazione del metodo volta a diminuirne la complessità computazionale e di memoria, e si discutono possibili strategie per un futuro miglioramento.

Keywords: *3D Shape comparison, Hybrid geometry-colour shape description, 3D Shape classification*

CALCOLO DI DESCRITTORI IBRIDI GEOMETRIA-COLORE PER L'ANALISI DI SIMILARITÀ DI FORME 3D

Andrea Raffo, Silvia Biasotti

01 Giugno 2016

Abstract

Il riconoscimento e la classificazione di forme geometriche hanno molteplici applicazioni in grafica computazionale, spaziando dallo sviluppo di search engine all'identificazione di anomalie in modelli medicali orientati al paziente, al design di nuovi farmaci e alla sperimentazione in silico. In tutti questi contesti, il confronto tra forme viene tradotto nel confronto tra descrizioni dell'oggetto. È quindi di fondamentale importanza che una buona descrizione dell'oggetto sia quanto più possibile completa, per risultare univoca, e allo stesso tempo concisa, per poter essere facilmente computabile e confrontabile con altre descrizioni. Inoltre, la descrizione geometrica di un oggetto 3D può essere corredata da informazioni colorimetriche, eventuali annotazioni e/o dati derivanti dalla sua acquisizione/provenienza. Nell'ambito generale del confronto di forme 3D, in questo lavoro si analizza un metodo che utilizza proprietà geometriche e colorimetriche. Rispetto alla implementazione presente nella letteratura, si studia una ottimizzazione del metodo volta a diminuirne la complessità computazionale e di memoria, e si discutono possibili strategie per un futuro miglioramento.

1. Introduzione

Per consentire al calcolatore di analizzare e descrivere un modello, l'oggetto deve essere opportunamente rappresentato in forma digitale (la cosiddetta rappresentazione geometrica, per esempio una mesh triangolare tridimensionale). Per la creazione di modelli 3D si può fare ricorso all'acquisizione di dati reali (attualmente esistono in commercio diversi tipi di scanner 3D, facili da usare ed economici, che permettono di costruire mesh triangolari), oppure si possono costruire virtualmente attraverso l'utilizzo di software di modellazione geometrica (3D Studio MAX¹, OpenSCAD² e Blender³ sono alcuni esempi di software molto utilizzati per la creazione di dati 3D).

Una volta che si ha a disposizione la rappresentazione geometrica dell'oggetto, per consentirne il confronto con altri modelli, è necessario estrarne una descrizione della forma che sia efficace, ossia codifichi le proprietà salienti della forma, ed efficiente, ossia sia ragionevolmente veloce da estrarre. Ottenute una o più descrizioni di una forma 3D è poi necessario definire opportune distanze (ad esempio metriche). A questo punto, essendo computazionalmente troppo complesso confrontare direttamente le rappresentazioni geometriche tra di loro, il confronto tra forme geometriche si traduce quindi nel calcolo delle distanze (metriche) tra coppie di descrizioni geometriche (i cosiddetti *shape descriptor*, o fingerprint, signature).

¹ <http://www.autodesk.com/products/3ds-max/overview>

² <http://www.openscad.org/>

³ <https://www.blender.org/>

Intuitivamente, i descrittori codificano "insiemi di proprietà" volte a sintetizzare una o più caratteristiche di un dato oggetto (ad esempio: il numero di lati, l'area, il genere, ... nelle figure poligonali piane). In generale, l'oggetto non è univocamente e interamente identificato dai suoi descrittori: essi devono essere sufficientemente selettivi da permettere discriminazioni (per esempio: il numero di lati non basta a identificare un poligono, ma se si unisce questa informazione agli angoli si può distinguere un rettangolo da un trapezio).

In questo lavoro analizziamo il metodo di confronto ibrido (in quanto combina informazioni geometriche e colorimetriche) proposto in [1] basato su spin images a supporto globale e ne discutiamo una possibile modifica volta a semplificarne la complessità computazionale.

Il metodo originale prevede la costruzione di un descrittore di forma e uno di colore per ciascun punto della mesh, la computazione di rispettive misure di similarità e la loro successiva combinazione mediante logica fuzzy. L'algoritmo opportunamente modificato nella codifica del descrittore e nella metrica usata per la calcolare la distanza tra due descrittori è stato testato su due dataset: il benchmark SHREC [2], costruito per la valutazione di metodi di retrieval, e il dataset HAMPSON [3], proveniente dalla scansione di oggetti 3D reali. I risultati di SHREC sono stati analizzati alla luce di tecniche diverse (precision-recall, la funzione di classificazione usata in SHREC2014 e diverse tecniche di clustering).

Il resto del report è organizzato come segue. In Sezione 2 presentiamo una panoramica dei metodi di retrieval di oggetti 3D che combinano informazione geometrica e colorimetrica, in Sezione 3 descriviamo in dettaglio il metodo implementato e le modifiche apportate mentre in Sezione 4 presentiamo i risultati di retrieval e classificazione che abbiamo ottenuto su 2 dataset, già utilizzati in letteratura per il confronto di mesh 3D con informazione colorimetrica associata. Infine, la Sezione 5 conclude il report e discute possibili sviluppi futuri e ulteriori modifiche al metodo.

2. Stato dell'arte

Le informazioni di tipo colorimetrico, benché siano comunemente combinate a quelle geometriche nei settori dell'object retrieval [4] e del geometry processing [5,6], non sono state utilizzate dalla maggior parte dei metodi 3D di object retrieval [7,8].

I primi tentativi di ideare descrittori 3D per oggetti dotati di texture sono basati su una descrizione feature-vector combinata con informazioni colorimetriche. In questo caso il colore è trattato come una proprietà generale senza considerare la sua distribuzione sulla forma. Un esempio di tale approccio è quello seguito da Suzuki [9] che codifica geometria e colore nei coefficienti di un vettore. Analogamente, Ruiz et al. [11] combinano la similarità geometrica basata sulla distribuzione di forma [12] con la similarità colorimetrica computata mediante il confronto degli istogrammi di distribuzione del colore, mentre in Starck e Hilton [13] le informazioni colorimetriche e di forma 3D sono concatenate in un istogramma.

Nel campo del riconoscimento di immagini, una strategia molto comune è quella di considerare delle patch locali che descrivono il comportamento della texture intorno ad un gruppo di pixel. Esempi sono i local binary pattern (LBP) [14], la scale-invariant feature transform (SIFT) [15], l'istogramma dei gradienti orientati (histogram of oriented gradients, HoG) [16] e le spin image [17]. La generalizzazione di questi descrittori a modelli con texture 3D è stata esplorata in diverse applicazioni, come nella VIP description [18], nel descrittore meshHOG [19] e nelle spin image con texture [20,1]. Ulteriori esempi sono le feature di colore CHLAC computate su dati di voxel 3D

proposti da Kanezaki et al. [21]; il metodo di campionamento introdotto da Liu et al. [22] per selezionare punti nelle regioni di alta variazione geometrica o di colore e definire una signature basata su istogrammi delle caratteristiche valutate in tali punti; il descrittore CSHOT [23], ideato per risolvere problemi di confronto di superfici basate su feature locali, come per esempio le corrispondenze punto a punto.

Un altro aspetto utilizzato per la caratterizzazione di proprietà locali e globali della forma è la simmetria [24]. Per esempio, Kazhdan et al. introducono il descrittore spherical harmonics per codificare la forma in accordo con la sua simmetria rotazionale intorno agli assi centrati nel centro di massa. In [25], tale descrittore è stato proposto in combinazione con descrittori colorimetrici al fine di analizzare modelli 3D con texture. Giachetti e Lovato [26] hanno introdotto la Multiscale Area Projection Transform (MAPT) per accoppiare il grado locale di simmetria radiale (in un dato range di scala) con una nozione legata all'alta simmetria shape, seguendo un approccio simile alla fast radial symmetry [27] usata in image processing. Variazioni pesate sul colore della trasformata MAPT, che combinano informazioni geometriche e colorimetriche, sono state presentate in [25,28].

Nel corso degli anni, grande attenzione è stata rivolta al confronto e recupero di forme 3D non rigide in quanto capaci di codificare la forma di oggetti in movimento e riconoscerli all'interno di scene più complesse. Al tal fine è necessario adottare descrittori di forma che siano invarianti a deformazioni isometriche di forma. Una metrica adatta per comparare forme non rigide è la geodetica; esempi di descrittori di forma tridimensionali basati su geodetiche sono le matrici di distanza geodetica [29,31] o i geodesic skeleton paths [30,31]. Oltre alle geodetiche, sono possibili scelte più sofisticate, come le diffusion e commute-time distanze [32]. Sulla base del fatto che queste distanze sono ben approssimate dall'operatore di Laplace-Beltrami, sono state proposte diverse descrizioni spettrali per caratterizzare le feature geometriche di forme tridimensionali non rigide [33], come la ShapeDNA [34], la heat kernel signature [35,36], la wave kernel signature [37], la global point signature [38] e la spectral graph wavelet signature [39].

Nel contesto di mesh tridimensionali dotate di texture, la photometric heat kernel signature [40-42] fonde geometria e colore in un descrittore locale-globale. L'idea alla base è quella di usare la struttura di diffusione per integrare la forma in uno spazio di alte dimensioni dove le coordinate embedding rappresentano le informazioni fotometriche. Seguendo la stessa idea, in [43] gli autori hanno generalizzato la distanza geodetica come una descrizione ibrida di forma capace di accoppiare informazioni di geometria e texture.

Altre classi invarianti possono essere interessanti nelle applicazioni, possibilmente dotate di trasformazioni non isometriche come deformazioni topologiche o scaling locale/globale. In questo caso, approcci topologici [44,45] offrono una struttura modulare nella quale è possibile collegare in proprietà multiple di forma come funzioni reali, così da descrivere forme e misurare la loro (dis)similarità rispetto a diverse nozioni di invarianza. Esempi di tali approcci sono i grafi di Reeb [46,47], le size function [48], i diagrammi di persistenza [49,50] e gli spazi di persistenza [51]. Recentemente, descrittori topologici hanno mostrato di essere una buona opzione per comparare forme dotate di informazioni colorimetriche [44].

Uno studio comparativo su alcuni metodi è stato proposto in [52]. In tale lavoro, sei metodi per retrieval e classificazione di forme tridimensionali dotate di texture sono stati implementati e confrontati sulla base di un dataset comune:

- i. Istogrammi ATP (area projection transform), dati colorimetrici e istogrammi congiunti di dati MAPT e RGB, basati su una descrizione geometrica multi scala capace di catturare simmetrie locali e globali accoppiate da istogrammi di canali normalizzati RGB
- ii. Codifica della geometria spettrale della forma geometrica con colore, combinanti un descrittore spettrale geometrico con un istogramma concatenato dei valori RGB del colore definito sui vertici del modello
- iii. Descrittori colore+forma, adottanti combinazioni (con pesi variabili) dell'istogramma dei valori RGB con un descrittore geometrico rappresentato da autovalori della matrice della distanza geodetica
- iv. Distribuzione dei canali del colore con persistenza omologica, combinanti diversi descrittori geometrici, colorimetrici ed ibridi, cioè il descrittore armonica sferica, le distribuzioni di forma delle distanze geodesiche pesate con gli attributi colorimetrici e una descrizione basata sulla persistenza e sullo spazio CIELab
- v. Rappresentazione multi-risoluzione delle informazioni geometriche e colorimetriche attraverso la combinazione di un approccio multi-visuale con Local Binary Pattern (LBP) combinati con istogrammi concatenati dei canali di colori CIELab
- vi. PHOG (photometric and geometric functions for textured shape retrieval), un metodo che combina un descrittore di forma basato su funzioni geometriche, un descrittore basato su persistenza e un descrittore puramente colorimetrico basato su spazio CIELab.

In questo lavoro utilizziamo alcuni dei test fatti in [52] per comparare il metodo implementato sullo stesso dataset e un altro insieme di dati provenienti da scansioni di beni culturali.

3. Descrizione del metodo

3.1 L'algoritmo originale [1]

Seguendo un approccio abbastanza comune, si costruiscono, per ogni modello, un set di descrittori che verranno poi utilizzati per calcolare le misure di similarità tra i vari oggetti. Più precisamente, per ogni modello sono calcolati un descrittore geometrico e separatamente vari descrittori di texture (il loro numero verrà precisato in seguito). Le informazioni contenute in tali codifiche saranno poi estratte per misurare separatamente la similarità di forma e di texture. Infine, da tali risultati si ottiene, utilizzando la logica fuzzy, una singola misura di similarità.

Di seguito, il metodo è descritto identificandone i sette passi salienti.

1. CARICAMENTO DEL MODELLO

Ogni modello deve essere opportunamente caricato, memorizzandone le informazioni relative alla struttura geometrica e alla colorazione (caricamento in memoria della rappresentazione geometrica).

L'approccio matematico su cui si fonda la schematizzazione di oggetti 3D è il seguente: si scompone uno spazio topologico in singole componenti facili da studiare (per esempio: un poliedro è scomponibile in facce, lati e vertici; una superficie è approssimabile con triangoli). In figura 1 si riporta un esempio di modello la cui discretizzazione è stata ottenuta tramite

complessi simpliciali (mesh triangolari, utili per la discretizzazione di superfici).

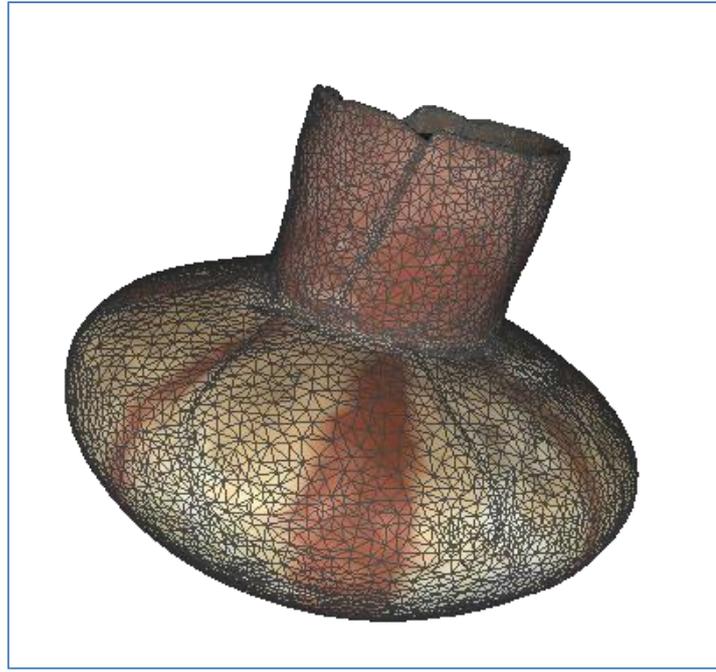


Figura 1 Triangolazione su un vaso antico, caratterizzata da 12681 vertici, 37676 lati e 24996 triangoli

Nel dettaglio, vengono caricate da un file COFF (Colored Object File Format) le liste dei vertici (identificati dalla terna delle coordinate), dei lati e dei triangoli (identificati rispettivamente da una coppia e da una terna di indici, ciascuno dei quali rappresenta un dato vertice).

La texture di ciascun oggetto si introduce assegnando ad ogni vertice una terna di valori (tra 0 e 255) rappresentanti le quantità dei colori primari rispetto alla terna RGB.

2. PRE-PROCESSING DEL MODELLO

Due operazioni sono eseguite prima della costruzione dei descrittori:

- Remeshing, cioè un ricampionamento dei vertici al fine di averne una distribuzione uniforme
- Rescaling, cioè una eventuale scalatura del modello. Tale operazione non è sempre una scelta ottimale, ad esempio un rescaling non permette di distinguere oggetti di diverse dimensioni.

3. COSTRUZIONE DELLE SPIN IMAGE PER SHAPE DESCRIPTOR

Lo shape descriptor si basa sulla costruzione delle *spin image*, di cui per chiarezza espongo una rapida descrizione.

Dato un sistema di riferimento in R^3 , si definisce punto orientato la coppia formata dal vettore \mathbf{p} , individuante un dato vertice (3D), e dal vettore normale \mathbf{n} alla superficie in \mathbf{p} . Per ogni punto dello spazio, individuato dal vettore \mathbf{x} , si può allora associare una coppia (α, β) di numeri reali rispetto al sistema di coordinate (\mathbf{p}, \mathbf{n}) , decomponendo \mathbf{x} lungo \mathbf{p} ed \mathbf{n} . Nella pratica si proietta il modello tridimensionale su un piano mediante l'espressione:

$$(\alpha, \beta) = \left(\sqrt{\|\mathbf{x} - \mathbf{p}\|^2 - (\mathbf{n} \cdot (\mathbf{x} - \mathbf{p}))^2}, (\mathbf{n} \cdot (\mathbf{x} - \mathbf{p})) \right)$$

La proiezione ottenuta è nota come spin map. Facendo variare \mathbf{p} per ogni vertice otterrò tante spin map quanti sono i vertici. Una spin image consiste in una discretizzazione della

proiezione appena ottenuta: lavorando in aritmetica finita non è infatti pensabile di immagazzinare tutte le informazioni fornite da una proiezione.

Si procede dunque nel seguente modo:

- si cerca una regione T di piano nella quale sia contenuta la proiezione (finita in quanto la superficie 3D è limitata), per esempio di forma rettangolare;
- si sceglie in numero N_a di livelli di discretizzazione di α e N_b di β (da notare che tale scelta andrà ad incidere sulla "bontà" della classificazione)
- si partiziona la regione T in $N_a \times N_b$ celle
- si alloca una matrice H a entrate nulle di dimensione $N_a \times N_b$: ad ogni entrata corrisponderà in modo biunivoco una cella della partizione di T
- data una certa spin map, si controlla in quale cella cade la proiezione di ciascuno dei vertici: in base a tale risultato, si incrementa ogni volta di 1 l'entrata della matrice H che corrisponde alla rispettiva cella.

La matrice ottenuta è quindi la spin image desiderata.

Per mostrare l'importanza di una "buona" discretizzazione, in figura 2 si mostrano le spin image ottenute a partire dal modello di un occhiali usando diverse discretizzazioni, ma partendo dalla stessa spin map.

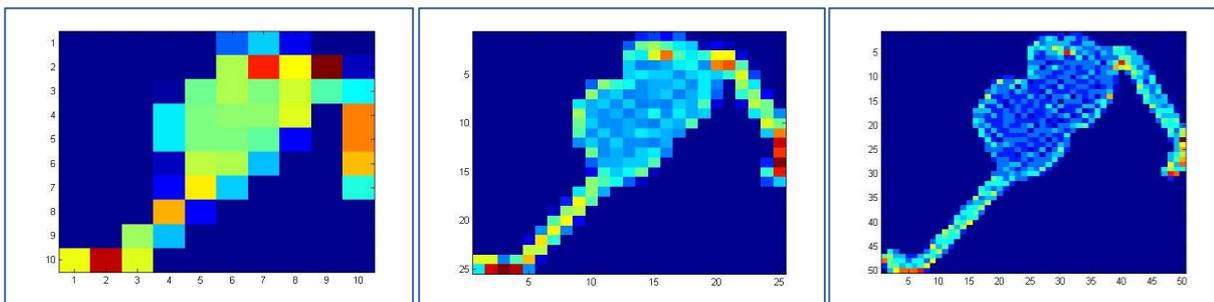


Figura 2 Confronto tra diverse spin image con $N_a=N_b=10,25,50$

4. COSTRUZIONE DELLO SHAPE DESCRIPTOR

Supponiamo che il modello caricato abbia v vertici, e sia $d:=N_a \times N_b$.

La costruzione del descrittore di forma segue i passi seguenti:

- si legge ognuna delle v spin image in ordine di riga (cioè si vettorizza la matrice riga per riga ottenendo i vettori s_i di dimensione d , con $i=1, \dots, v$);
- si definisce la matrice $d \times v$ $S := [s_1, \dots, s_v]$ intendendo i vettori creati al passo precedenti disposti per colonna;
- Si costruisce un vettore colonna $d \times 1$ S^{medio} ponendo come entrata i -esima la somma degli elementi della i -esima riga di S (cioè il valor medio della i -esima riga a meno di costante di proporzionalità);
- si sottrae ad ogni colonna di S il vettore S^{medio} ottenendo la matrice S^- ;
- si esegue una PCA (*Principal Components Analysis*) diagonalizzando la matrice $(S^-)^T (S^-)$ di dimensione $d \times d$: ciò è possibile in quanto quadrata, simmetrica e reale;
- la matrice $E_{sh} = [e_1, \dots, e_d]$ avente come colonne gli autovettori ottenuti dalla decomposizione è lo shape descriptor desiderato, ed è nota come *matrice degli eigen spin images*; ad esso è associato il vettore $V_{sh} = [\lambda_1, \dots, \lambda_d]$ contenente i relativi autovalori.

L'articolo originale propone, al fine di ridurre la dimensione del descrittore, di considerare un numero ridotto $c_{sh} < d$ di autovettori corrispondenti agli autovalori più grandi in modulo: dalla teoria dell'analisi in componenti principali risulta infatti come un maggior valore (in modulo degli autovalori) sia correlato ad una maggiore rappresentatività dell'autovettore associato.

5. COSTRUZIONE DEI COLOR DESCRIPTOR

Per la realizzazione dei descrittori di colore si suppone che a ogni vertice sia associata una terna RGB. Si discretizza, sempre per ragioni di aritmetica finita, lo spazio colore in l livelli. L'idea base su cui si fonda questa parte è la seguente, ripetuta per ogni vertice di riferimento:

- o parallelamente alla costruzione dello shape descriptor, si allocano a zero l matrici $N_a \times N_b$ (ciascuna delle quali rappresenta uno degli l livelli);
- o per ogni vertice, si cercano gli indici di riga i^* e colonna j^* corrispondenti all'entrata incrementata della matrice H dello shape descriptor;
- o si controlla a quale degli l livelli di quantizzazione del colore appartiene il vertice in esame: si andrà ad incrementare l'entrata (i^*, j^*) della corrispondente matrice.

Con tale costruzione, la spin image della forma sarà la somma delle l spin image del colore. A questo punto si costruiscono l descrittori di colore analogamente a quanto fatto per quello della forma (cioè tramite la PCA- decomposizione autovalori/autovettori e successiva riduzione del loro numero).

I descrittori di colore relativi a un modello possono essere espressi dalla coppia (C, V_{col}) :

$$C = \{E_1, \dots, E_l\}$$

$$V_{col} = \{V_1, \dots, V_l\}$$

dove:

$$E_i = [e_{i1}, \dots, e_{ici}], V_i = [\lambda_{i1}, \dots, \lambda_{ici}] \text{ e } c_i = \text{cut-off del livello } i\text{-esimo.}$$

Ma come quantizzare lo spazio colore?

Il metodo adottato funziona nel modo seguente: prima di tutto, i colori dei vertici sono convertiti nello spazio di colore CIELAB [53] rappresentato in figura 3 (uno spazio dotato di metrica euclidea che "rispecchia il nostro concetto di vicinanza/lontananza dei vari colori"):

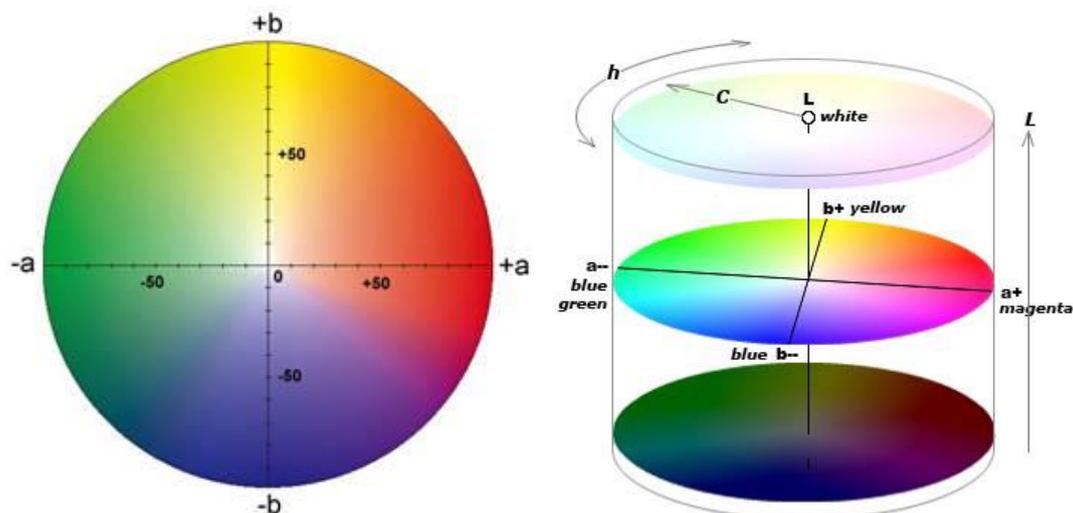


Figura 3 Spazio CIELAB

Il modello CIELAB mappa lo spazio tridimensionale RGB, in un altro spazio con un sistema di coordinate cilindriche come mostrato in figura 3. La componente L denota la luminosità del colore, mentre le componenti a e b rappresentano insieme le informazioni cromatiche. L'articolo propone di suddividere tale spazio in 6 regioni differenti, come mostrato nella tabella 1.

Tabella 1: Intervalli di quantizzazione del colore

| Quantization level | Luminance (L) | Hue angle (h_{ab}) | Color descriptor |
|--------------------|---------------|---------------------------|----------------------------|
| 1 | $L < 10$ | \forall | "very dark" |
| 2 | $10 < L < 98$ | $0.35 < \vartheta < 1.4$ | red, purple, pink, magenta |
| 3 | $10 < L < 98$ | $1.4 < \vartheta < 1.92$ | green, lime |
| 4 | $10 < L < 98$ | $1.92 < \vartheta < 3.42$ | blue, aqua, turquoise |
| 5 | $10 < L < 98$ | $3.42 < \vartheta < 5.93$ | yellow, olive, orange |
| 6 | $L > 98$ | \forall | "very bright" |

La prima regione include tutti i colori scuri (quelli con luminosità < 10), mentre l'ultima contiene i chiari ($L > 98$). La parte centrale è divisa ulteriormente in 4 sottoregioni sulla base dello *hue angle* h_{ab} :

$$h_{ab} := \arctg\left(\frac{b}{a}\right).$$

6. MISURA DI SIMILARITA'

Il passo finale consiste nell'elaborazione della misura di similarità sulla base dei descrittori introdotti nei punti precedenti. Consideriamo un *query model* \mathcal{M}_q e $\mathcal{C} = \{\mathcal{M}_n: n=1,2,\dots,N\}$ l'insieme di tutti i modelli candidati.

E' necessario confrontare \mathcal{M}_q e ogni \mathcal{M}_n con $n=1,\dots,N$ usando i descrittori precedentemente realizzati. A tal fine, nell'approccio proposto, saranno prima determinate separatamente le misure di similarità di forma e di colore per ciascuna coppia di modelli; successivamente, le due misure saranno poi combinate assieme usando la logica fuzzy.

o Shape comparison

Per avere una visione più chiara possibile, iniziamo a discutere il caso della misura di similarità per la forma. Siano dunque $E = [\mathbf{e}_1, \dots, \mathbf{e}_{c_{sh}}]$ la matrice degli eigen-spin images, $\mathbf{V}_{sh} = [\lambda_1, \dots, \lambda_{c_{sh}}]$ il vettore dei relativi autovalori, c_{sh} il valore di cut-off degli autovalori-autovettori, N il numero di modelli. La misura di similarità tra il query model \mathcal{M}_q e il modello n -esimo \mathcal{M}_n è data dall'espressione:

$$Sim_{sh}(M_q, M_n) = \frac{\sum_{i=1}^{c_{off}} w(\lambda_{i,q}) |\rho(\mathbf{e}_{i,q}, \mathbf{e}_{i,n})|}{\sum_{i=1}^{c_{off}} w(\lambda_{i,q})}$$

dove:

$$c_{off} = \min(c_{sh,q}, c_{sh,n})$$

è il minimo tra i valori di cut-off del query model e del candidato n-esimo. Il simbolo ρ rappresenta il coefficiente di correlazione di Pearson mentre w è la composizione tra modulo e potenza:

$$w: [0, +\infty) \rightarrow [0, +\infty) \\ x \rightarrow |x|^\vartheta$$

con $0 < \vartheta < 1$.

Si osservi come la misura di similarità così definita non sia simmetrica, cioè:

$$Sim_{sh}(M_a, M_b) \neq Sim_{sh}(M_b, M_a)$$

○ Color comparison

L'obiettivo della procedura di confronto tra modelli è di valutare la similarità delle differenti regioni di colore, al fine di riconoscere distribuzioni simili sulla loro forma. L'idea proposta è di confrontare ogni matrice (eigen-spin image) del modello query corrispondente a un dato livello di quantizzazione con quelle degli altri livelli del modello candidato. La ragione di tale procedura è legata al fatto che le eigen spin image di regioni con differenti colori ma distribuzione simile sulla loro forma sono altamente correlate.

Vediamo nel dettaglio come viene definita in questo caso la similarità tra modelli. Al fine di confrontare i descrittori di colore del modello query \mathcal{M}_q con il candidato n-esimo \mathcal{M}_n , considero la massima similarità tra l'i-esima matrice eigen-spin image $E_{i,q}$ del query descriptor e tutte le matrici dei candidati $E_{j,n}$, con $j=1, \dots, l$.

La misura di similarità per l'i-esimo livello di quantizzazione è cioè definita come:

$$Sim_{col}^i(M_q, M_n) = \max_{1 \leq j \leq l} \left\{ \frac{\sum_{k=1}^{c_{ij}} w(\lambda_{ik,q}) |\rho(e_{ik,q}, e_{jk,n})|}{\sum_{k=1}^{c_{ij}} w(\lambda_{ik,q})} \right\}$$

dove:

$$c_{ij} = \min(c_{i,q}, c_{j,n})$$

è il minimo tra i valori di cut-off.

Definendo ora:

$$w_{li} = \sum_{k=1}^{c_{i,q}} |\lambda_{ik,q}|^\vartheta$$

con $0 < \vartheta < 1$, è possibile pesare i vari livelli di quantizzazione e ottenere la misura di similarità (di nuovo non simmetrica) colore tra il modello query e quello n-esimo:

$$Sim_{col}(M_q, M_n) = \frac{1}{\sum_{i=1}^l w_{li}} \sum_{i=1}^l w_{li} Sim_{col}^i(M_q, M_n)$$

7. AGGREGAZIONE FUZZY

Infine, la distanza tra due oggetti è data dalla combinazione delle misure di similarità geometrica rispetto al colore. A questo riguardo, l'articolo [1] propone uno studio statistico sui pesi da associare a ciascuna misura di similarità basata sulla logica fuzzy.

3.2 Modifiche apportate all'algoritmo

All' algoritmo sopra descritto sono state apportate diverse modifiche, principalmente per avere un consistente risparmio in termini di costo computazionale e di memoria e quindi per rendere possibile l' applicazione del metodo a dataset composti da centinaia di modelli. Inoltre, si è risolto un problema concettuale legato alla rettangolarità della matrice e, soprattutto, introdotta una misura di similarità simmetrica, ossia tale che, dati due oggetti A e B, $d(A,B)=d(B,A)$.

- Codifica della texture

Per il tipo di descrittore scelto, è importante che il colore sia associato a ciascun vertice della rappresentazione geometrica. Nello sviluppo dell' algoritmo si è quindi memorizzato il colore in ciascun vertice della triangolazione e sono modificate alcune librerie per il caricamento dei dati (IMATISTL).

- Campionamento adattivo e uniforme dei vertici di riferimento

L' algoritmo presentato nell' articolo prevedeva che, fissato un dato modello tridimensionale, venisse effettuata una proiezione piana per ogni vertice della triangolazione (vedi punto 3 del capitolo precedente). Tale approccio ha il pregio di considerare tutte le informazioni geometriche provenienti dai diversi punti di vista offerti al variare del vertice di riferimento, ma risulta computazionalmente troppo oneroso in quanto il loro numero può arrivare a diverse decine di migliaia.

Si è pensato allora di considerarne solo un numero ridotto (ad esempio 30), con l' obiettivo di coniugare il miglioramento delle prestazioni a una buona omogeneità di campionamento: a tal fine è stata implementata una variante dell' algoritmo dei cammini minimi di Dijkstra. In teoria dei grafi un *cammino minimo* tra due vertici è un percorso, non sempre univocamente determinato, che collega tali vertici e che minimizza la somma dei costi associati. Tale strumento consente, con opportune modifiche, di partire da un dato vertice e trovare quello a lui più distante rispetto al cammino minimo; se iterato, consente di calcolare i vertici più distanti da un insieme fissato di vertici.

Di seguito sono riportati i passi dell' adattamento dell' algoritmo di Dijkstra ai fini di sottocampionamento.

- *Associazione del grafo alla triangolazione*: i vertici del modello in esame sono numerati utilizzando i naturali a partire dallo 0. Ad essi è stato poi associato un grafo costruendo gli archi in base alla triangolazione caricata e associando a ciascuno di essi la distanza euclidea tra i rispettivi vertici.
- *Prima esecuzione dell' algoritmo*: partendo da un vertice di riferimento scelto (per comodità si è preso il primo della lista caricata), si esegue l' algoritmo di Dijkstra ottenendo quindi il vertice che si trova a "distanza maggiore" da quello di partenza (nel senso del massimo cammino minimo).
- *Campionamento*: partendo dai due vertici ottenuti ai passi precedenti, si riapplica n-2 volte l' algoritmo, ottenendone così n. In particolare, l' i-esimo vertice campionato dovrà avere "distanza massima" dagli i-1 precedentemente trovati.

Si osserva che la riduzione del numero di vertici di riferimento comporta una diminuzione della dimensione di S, ma non di quella del descrittore shape.

- Principal Component Analysis (PCA)

L'applicazione della PCA per la misurazione della similarità è già stata descritta nei passi dell'algoritmo, ma in realtà è stata introdotta dopo uno studio dell'articolo che ha mostrato diverse ambiguità: nello scritto si parlava infatti di effettuare uno studio degli autovettori della matrice S rettangolare, il che ha portato al quesito su come poter conservare l'idea della decomposizione in accordo con i principi base della teoria della diagonalizzazione.

- Simmetrizzazione delle matrici di similarità

Come già detto, le matrici ottenute per la forma e il colore non risultano simmetriche: si ha infatti che la misura di similarità tra un modello A ed uno B risulta in generale diversa da quella tra B ed A . Tale situazione risulta insoddisfacente sia dal punto di vista matematico, che da quello modellistico del problema in quanto non è possibile ottimizzare la navigazione del dataset in base alla conoscenza pregressa.

Al fine di risolvere questo problema, si è scelto di simmetrizzare le matrici considerando la minima entrata tra le due situazioni possibili: l'uso del minimo consente di sostenere con certezza che due modelli abbiano almeno una certa percentuale di similarità rispetto alla modellizzazione considerata.

4. Esperimenti

4.1. Dataset

Per valutare il metodo sono stati considerati due dataset distinti: il primo creato ad hoc per lo studio di algoritmi di riconoscimento e il secondo proveniente dalla scansione 3D di oggetti reali.

Il benchmark SHREC[2] è nato per valutare la performance di metodi di retrieval e classificazione di oggetti forniti anche di una texture. A tal fine è composto da un dataset di oggetti pre-classificati e di alcuni strumenti di valutazione della performance. Tutti i metodi sono valutati sullo stesso dataset e con gli stessi strumenti. In figura 4 sono riportati i modelli facenti parte di questo benchmark.



Figura 4 Benchmark SHREC [2]

Il dataset è composto da 572 modelli sintetici, raggruppati in 16 classi geometriche. Ogni classe eccetto una è ottenuta a partire da 12 modelli base, corrispondenti a 4 forme di base a cui sono applicate 3 diverse texture. A ciascun modello base sono state applicate 2 trasformazioni, ottenute con la combinazione di una deformazione geometrica (che variano dal re-sampling, all'aggiunta di rumore Gaussiano, alla deformazione affine, non rigida e non isometrica) con una deformazione della texture (schiarimento/scurimento, deformazioni topologiche dei pattern, trasformazioni affini applicate ai canali RGB), fino ad ottenere 36 modelli per classe.

L'ultima classe contiene 32 oggetti, costruiti a partire da 8 modelli base e applicando a ciascuno di essi tre diverse trasformazioni.

Il dataset HAMPSON, utilizzato in [3], a differenza del benchmark SHREC, proviene dalla scansione 3D di oggetti reali. In questo caso si considera una collezione di mesh triangolari tridimensionali dotate di texture fornite dal "Virtual Hampson Museum".

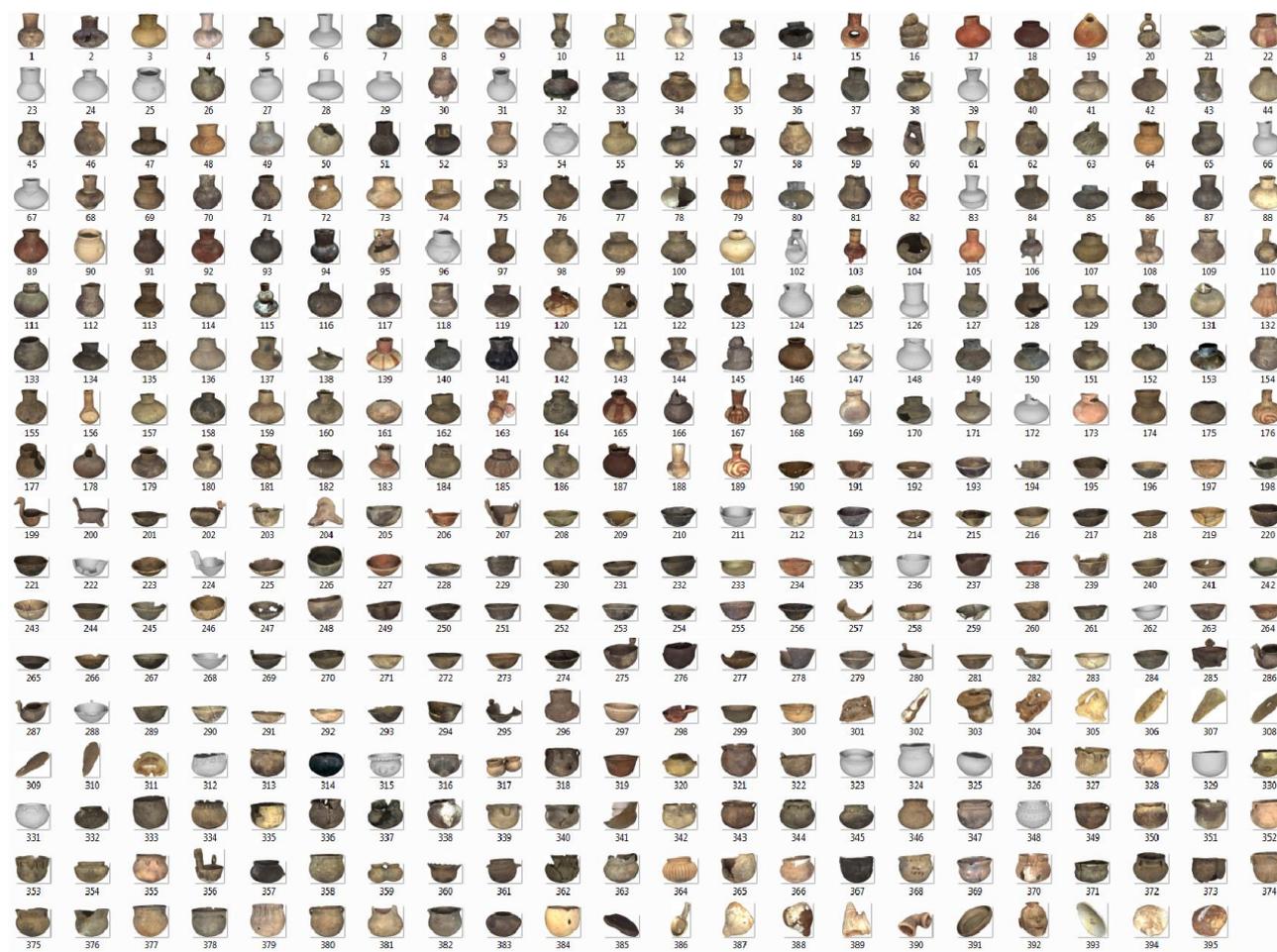


Figura 5 HAMPSON dataset (VHM, <http://hampson.cast.uark.edu>)

Il dataset, riportato in figura 5, è composto da 442 modelli, 395 dei quali disponibili per download, rappresentanti diversi manufatti di nativi americani risalenti a un periodo tra i 350 e i 600 anni fa. La maggior parte dei modelli è raggruppabile in 3 classi geometriche, note come "bottles" (189 elementi), "bowls" (112 elementi), and "jars" (73 elementi). Gruppi più piccoli sono comunque stati identificati dagli esperti del VHM, per un totale di 13 classi geometriche. Nessuna classificazione di texture è disponibile al momento, e alcuni modelli sono addirittura sprovvisti di colorazione oppure presentano imperfezioni nella triangolazione o nella texture. Diversi reperti sono comunque realizzati con lo stesso materiale, come ceramiche o pietra, oppure sono caratterizzati dai medesimi ornamenti. Tutte queste proprietà possono essere espresse in termini di informazioni fotometriche e conservate come informazioni nella texture.

4.2. Analisi dei risultati

I dataset a disposizione sono stati elaborati mediante l'algoritmo modificato, estraendo 30 punti di campionamento in ciascun modello (minimi cammini di Dijkstra) e scegliendo $N_a=N_b=10$ passi di discretizzazione della spin map (come suggerito dall'articolo originale).

I risultati ottenuti sul benchmark SHREC sono stati successivamente analizzati mediante diverse tecniche di classificazione:

1. PRECISION-RECALL

2. UTILIZZO DELLA FUNZIONE DI CLASSIFICAZIONE USATA IN SHREC 2014
3. CLUSTER ANALYSIS

Di seguito si riporta una breve descrizione dei tre esperimenti effettuati e degli output ottenuti:

4.2.1. PRECISION-RECALL (PR):

Precision e recall sono due misure statistiche molto utilizzate in diversi ambiti, tra cui l'information retrieval [54]. La *precision* è definita come il rapporto tra il numero di oggetti etichettati correttamente come appartenenti alla classe in esame e il numero totale degli elementi etichettati come suoi membri. Il *recall* è invece definito come il rapporto tra il numero di oggetti etichettati correttamente come appartenenti alla classe e il numero di elementi che ne sono in realtà membri.

Di seguito si riportano, per completezza, le formule di definizione:

$$Precision = \frac{\text{vero positivo}}{\text{vero positivo} + \text{falso positivo}}$$

$$Recall = \frac{\text{vero positivo}}{\text{vero positivo} + \text{falso negativo}}$$

Dalle definizioni segue che precision e recall assumono valori tra 0 ed 1.

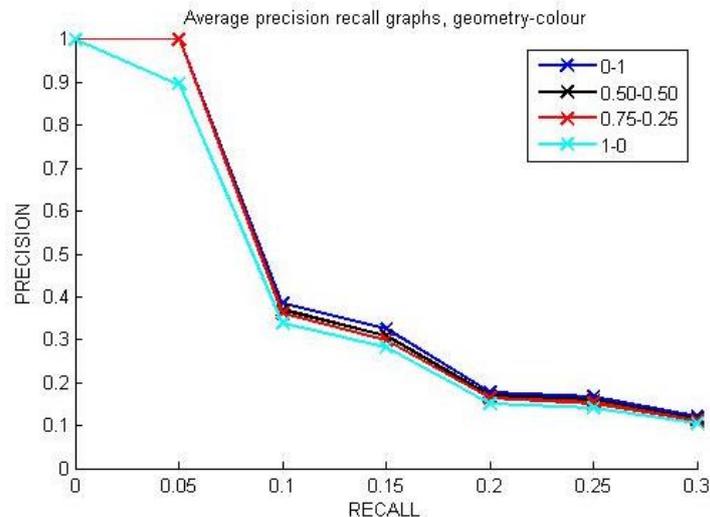


Figura 6 Precision Recall per la coppia di pesi geometria-colore

Si riportano in Figura 6 i risultati ottenuti sul benchmark SHREC dall' algoritmo modificato pesando in modi diversi la geometria e il colore (dove i pesi sono due reali non negativi di somma 1, riportati in legenda).

Dal grafico di Figura 6 emerge che, con le scelte effettuate, la precisione aumenta al crescere del peso associato al colore: all'aumento di tale peso, infatti, corrisponde un innalzamento delle ordinate. Tale risultato può essere spiegato dal fatto che in realtà anche considerando solo il colore si mantengono le informazioni geometriche: bisogna ricordare che la somma delle spin-image colore è proprio la spin image della forma geometrica. I descrittori del colore non sono quindi puramente

colorimetrici ma mantengono buona parte dell'informazione geometrica. Se confrontato con alcuni metodi classici della letteratura, il metodo mostra però una minore capacità di determinare la similarità tra i modelli, vedi

Figura 7.

Il grafico si riferisce al report ufficiale del benchmark del 2014[2].

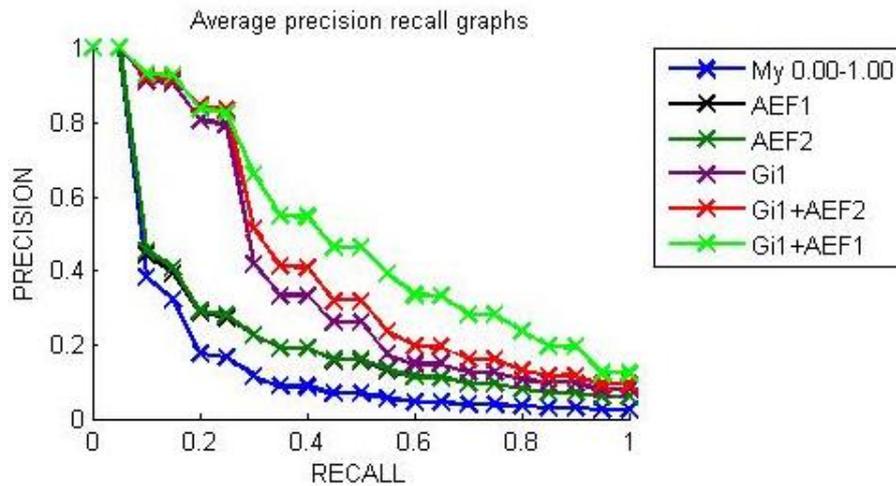


Figura 7 Confronto del metodo in esame con alcuni presenti in letteratura, le etichette dei metodi si riferiscono ai metodi come presentati in [2].

In figura 8 è riportato il precision-recall del dataset HAMPSON, ottenuto con le stesse discretizzazioni precedenti, nel caso di descrittore puramente geometrico: si ricorda infatti che questo dataset è privo di una classificazione che consideri la texture.

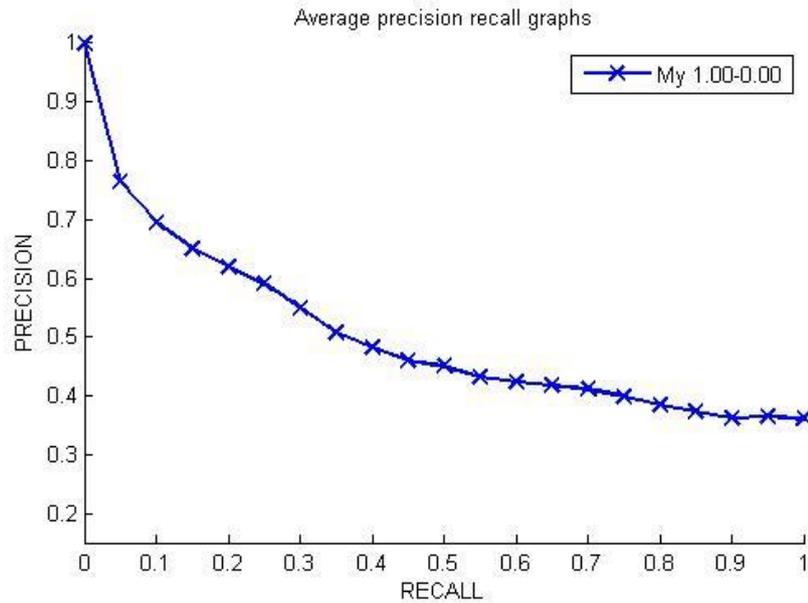


Figura 8 Precision Recall del dataset Hampson

Si noti, rispetto al benchmark, come i punti di campionamento (segnati con delle x) abbiano ordinate maggiori: questo fatto mostra come la classificazione ottenuta tramite il metodo implementato sia in effetti "più fedele" a quella nota rispetto a quanto non succeda in SHREC.

4.2.2. UTILIZZO DELLA FUNZIONE DI CLASSIFICAZIONE USATO IN SHREC 2014:

Il file Matlab valuta classificazione NN.mat, già precedentemente utilizzato per l'analisi del benchmark originale, permette, nota la classificazione (geometria + colore, nel caso in esame), di individuare il numero di successi e di assegnare un'etichetta ad ogni modello (consistente in un naturale a partire da 1).

Nel caso della classificazione SHREC, pesando 1 la geometria e 0 il colore, si hanno 354 successi su 572 (pari al 61.89%). Associando peso 0 alla geometria e 1 al colore, si ottengono, considerando la classificazione puramente colorimetrica nota per costruzione, un numero di successi di 103 su 572 (pari al 18,01%), risultato pressoché analogo a scegliere entrambi i pesi 0.5 (97 successi su 572).

Tre ulteriori misure di valutazione (nearest neighbour NN, first tier FT e second tier ST), che possono variare nell'intervallo [0,1], sono ulteriormente esaminate per verificare la frazione dei modelli recuperati che sono listati nei primi k elementi, dove k può essere 1 (NN), la dimensione della classe a cui appartiene il modello (FT) o il suo doppio (ST). In particolare, per una classe di n elementi k=1 indica il nearest neighbour (NN), k=n-1 il first tier e k=2(n-1) il second tier.

La tabella 2 riporta i risultati ottenuti sul dataset SHREC per le matrici di similarità assegnando pesi diversi a geometria e colore. Dai risultati ottenuti si evince che per questo dataset la performance ottenuta con le sole spin images "geometriche" è nettamente superiore alla combinazione con colore.

Tabella 2

| Pesi geometria-colore | NN | FT | ST |
|-----------------------|----|----|----|
|-----------------------|----|----|----|

| | | | |
|---------|----------|----------|----------|
| 1.0-0.0 | 0.617839 | 0.206129 | 0.302192 |
| 0.5-0.5 | 0.120370 | 0.156944 | 0.205165 |
| 0.0-1.0 | 0.138166 | 0.156944 | 0.208351 |

4.2.3. UTILIZZO DI UNA TECNICA DI CLUSTERING PER L'INDIVIDUAZIONE AUTOMATICA DI CLASSI DIFORME:

Al fine di raggruppare gli elementi in classi con caratteristiche "omogenee" si è pensato di utilizzare alcune tecniche di "clustering" che appartengono alla famiglia delle tecniche di statistica multivariata. A titolo di esempio si è presa in considerazione la matrice di similarità ottenuta con peso di colore pari a 1. Per poter applicare tali tecniche direttamente alle distanze in un dataset di forme geometriche non immerse in \mathbb{R}^n , come primo passo si è provveduto a una proiezione delle forme in uno spazio opportuno, in questo caso \mathbb{R}^2 . Per far questo si è utilizzato una tecnica basata sulla nozione di diffusion distance [55] precedentemente utilizzata in[56].

A partire dalla matrice di similarità M, calcoliamo la affinity matrix utilizzando un nucleo Gaussiano (sostituibile con altri kernel):

$$W_{ij} = \exp\left(-\frac{M_{ij}}{2\sigma^2}\right)$$

dove $\sigma=2$. L'uso dell'esponenziale permette di "avvicinare" punti a grande distanza.

Definendo la matrice diagonale D:

$$D_{ii} = \sum_j W_{ij}$$

è possibile ottenere la matrice normalizzata M:

$$M = D^{-1}W$$

M, per costruzione, può essere vista come una matrice stocastica: M_{ij} sarà la probabilità di transizione dallo stato i a quello j. La probabilità di transizione può essere interpretata anche come la "forza di connessione" tra due stati.

Si esegue infine una decomposizione autovalori-autovettori sulla matrice M, ottenendo gli autovalori $\lambda_0 = 1 > \lambda_1 \geq \dots \geq \lambda_{n-1} \geq 0$ ed autovettori $\psi_0, \dots, \psi_{n-1}$. Si considera dunque la matrice $\Psi = (\lambda_1^t \psi_1, \dots, \lambda_{n-1}^t \psi_{n-1})$: la riga i-esima corrisponde alle coordinate dell'i-esimo punto. Le coordinate cambiano a seconda del valore di t, il quale va a modificare il peso degli autovalori: nel mio lavoro è stato preso $t=3$.

Per visualizzarli graficamente, ci limitiamo a considerare le prime due colonne: si ottiene una proiezione su piano.

Tra le tecniche esistenti in letteratura, la matrice delle misure di similarità colore di SHREC è stata analizzata mediante la tecnica *DBSCAN (Density-Based Spacial Clustering of Applications with Noise)*[57] e Dominant Set [58].

- La tecnica DBSCAN è molto nota nella letteratura scientifica, connette regioni di punti ad alta densità. Necessita di due parametri: il numero minimo di punti richiesti per formare un cluster, e un valore ϵ per determinare l' ϵ -vicinato di un punto. Nel caso in cui i punti non siano collocati in modo chiaro in un cluster, viene associata una label apposita.

La tecnica del DBSCAN fornisce, nel caso SHREC, un gran numero di modelli non classificati in alcun cluster: in figura 9 tali punti sono rappresentati con il simbolo "+". Inoltre, nonostante siano identificati 12 clusters, ben 394 modelli vengono classificati come appartenenti ad un'unica classe (in figura 9, i punti gialli).

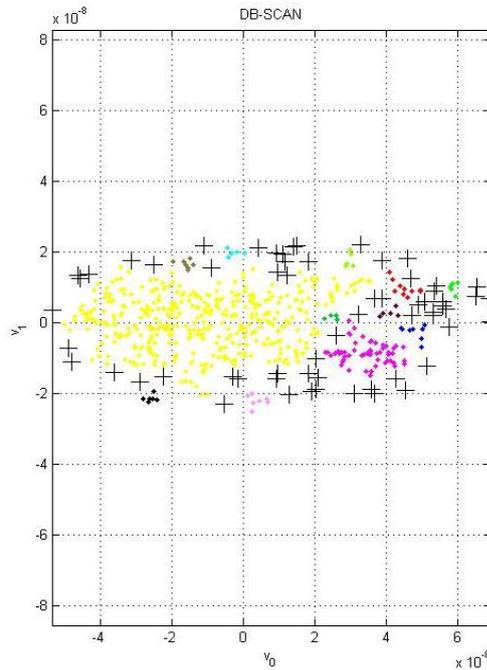


Figura 9 Output dell'algoritmo DBSCAN sulla matrice di similarità valutata per il dataset SHREC14

- **DOMINANT SET:** è una tecnica di clustering basata sulla teoria dei grafi, già utilizzata in [58]. Con questa tecnica, data la matrice M delle misure di similarità, l'obiettivo è quello di raggruppare oggetti sulla base delle relazioni codificate in M senza utilizzare nessuna immersione in \mathbb{R}^n degli oggetti del database: i membri di uno stesso raggruppamento dovranno essere caratterizzati da alta omogeneità, al contrario ci sarà grande disomogeneità tra le entità dentro e fuori da ciascun cluster. Ogni cluster viene rappresentato da un vettore reale n -dimensionale x , le cui componenti esprimono l'appartenenza di una componente al raggruppamento in esame: se il valore è alto/basso, l'associazione sarà forte/debole. Modelli non appartenenti al cluster in esame vengono rappresentati con 0.

La *coesione* degli elementi di un cluster è espressa tramite la forma quadratica $Q = \mathbf{x}^t \mathbf{M} \mathbf{x}$, cosicché il problema di partenza è equivalente alla massimizzazione di Q per $\mathbf{x} \in \Delta := \{\mathbf{x} \in \mathbb{R}^n | \mathbf{x} \geq 0 \text{ ed } e^t \mathbf{x} = 1\}$ di cui un *dominant set* corrisponde ad una soluzione. Per ottenere un dominant set è sufficiente utilizzare il seguente metodo iterativo:

$$x_i(t+1) = x_i(t) \frac{(M\mathbf{x})_i}{\mathbf{x}^t M \mathbf{x}}$$

dove il pedice i indica la componente del vettore n -dimensionale.

La soluzione può essere raggiunta per la proprietà di simmetria della matrice M . Il metodo porta quindi a un nuovo vettore che identifica un cluster consentendo di ridurre le dimensioni di M , e quindi di riapplicare il metodo fintanto che siano presenti modelli "isolati".

La tecnica del dominant set consente di ovviare ad alcune problematiche della tecnica precedente, senza però mostrare ancora risultati compatibili con la classificazione nota a priori: se da un lato elimina il problema di gruppi troppo numerosi (come nel caso del DBSCAN), dall'altro non riesce però ancora ad ottenere risultati corrispondenti a quelli attesi.

A titolo di esempio, in figura 10 si riporta la clusterizzazione ottenuta in 100 passi e 16 classi mediante la precedente immersione nello spazio bidimensionale. L'output rivela la difficoltà del metodo a identificare le classi in maniera "ottimale", anche se si osserva come i colori siano presenti in maniera più o meno omogenea.

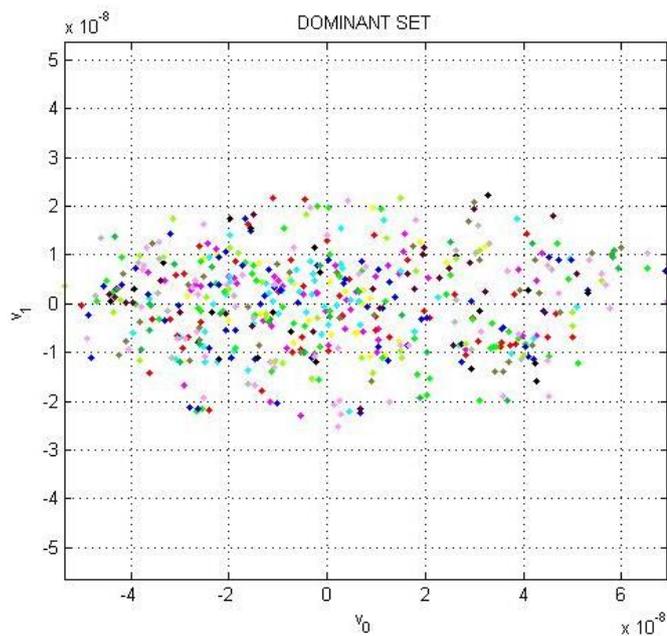


Figura 10 Dominant Set

I metodi statistici testati con SHREC offrono un buon approccio per inferire dalla matrice di similarità informazioni utili per classificare un insieme di modelli, ma mostrano però risultati insoddisfacenti.

5. Conclusioni

Lo studio del metodo e il suo confronto con i risultati di altri algoritmi hanno permesso di stilare una lista di possibili modifiche per un futuro miglioramento.

Come già evidenziato nella descrizione della costruzione dei descrittori, una limitazione fondamentale sta nell'eccessiva approssimazione con cui vengono discretizzate le spin image. Tale scelta è però da ricercarsi nell'eccessivo costo computazionale che il metodo richiede e nella grande quantità di memoria necessaria a codificare molte spin image di dimensione elevata. Malgrado l'ottimizzazione del calcolo del descrittore, il calcolo delle similarità tra tutti gli oggetti nel dataset ha richiesto un tempo di circa 6 giorni per il dataset SHREC e di circa 5 per HAMPSON con $N_a=N_b=10$, contro i rispettivi 24 e 20 giorni nel caso di $N_a=N_b=13$.

Si è osservato in particolare che, a variazioni del numero dei passi di discretizzazione delle spin image da 10 a 13 corrisponde un'alterazione della misura di similarità che in percentuale arriva a toccare il 25%: questo fenomeno è sintomo del fatto che l'approssimazione effettuata porta alla perdita di troppe informazioni forma-colore, il che riduce eccessivamente il numero di modelli correttamente individuati. Come mostrato dalla figura 2, infatti, i valori di N_a e N_b in gioco portano ad una pessima approssimazione del profilo.

Come possibile sviluppo riteniamo sia molto interessante cercare di ottimizzare il metodo per permettere una più fitta discretizzazione dei piani su cui proiettare le spin image che vada di pari passo ad una netta riduzione del costo computazionale della misura di similarità, eventualmente modificandola totalmente, ad esempio attingendo a misure di similarità tra immagini. Alla luce dei risultati ottenuti, è inoltre possibile applicare diversi metodi di clustering per individuare quello che meglio si adatta a un dataset. Questo approccio potrebbe essere accompagnato, per esempio, da una diversa codifica delle spin image e modificando ulteriormente la codifica delle informazioni in una matrice.

La libreria EIGEN qui utilizzata limita la dimensione delle spin image usate per la codifica della forma, permette di eseguire l'algoritmo con al più $N_a=N_b=13$: nel momento in cui una soluzione al crescente costo computazionale sia trovata, si può pensare di ovviare al problema tramite la libreria LAPACK (meno user-friendly ma più potente dal punto di vista numerico).

Un'altra possibile variazione riguarda il metodo di campionamento dei vertici del modello: come già spiegato non è computazionalmente sostenibile l'elaborazione delle spin image per ogni vertice, ma occorre determinarne un sottoinsieme sufficientemente piccolo che conservi però abbastanza informazioni da poter riassumere il modello stesso. Una possibilità può essere, ad esempio, quella di valutare cosa accade scegliendo i punti proiettandoli tramite un cilindro contenente l'oggetto studiato.

Dal punto di vista dello spazio colore CIELAB, potrebbe essere preferibile cambiare il numero di livelli di discretizzazione qualora i modelli considerati presentino per esempio un ampio range di variabilità. Occorre però rimarcare il fatto che ogni livello aggiuntivo comporta la creazione di un'ulteriore matrice per ogni vertice di campionamento (e, quindi, è necessario trovare un compromesso tra una crescente precisione e la sostenibilità computazionale).

Per il confronto dei descrittori ottenuti con questo metodo, infine, sarebbe opportuno investigare mediante l'utilizzo del paradigma Bag of Features (anche noto come Bag of Words, BoW, una tecnica di classificazione molto usata in information retrieval e nell'object recognition) in quanto ogni oggetto è descritto da una sequenza di immagini (features).

6. Ringraziamenti

Desidero ringraziare, innanzitutto, la prof. Emanuela Sasso del Dipartimento di Matematica dell'Università degli Studi di Genova, grazie a cui ho avuto l'opportunità di preparare la documentazione necessaria al mio tirocinio in tempi brevi. Questa esperienza non sarebbe stata possibile senza la dedizione mostrata dalla dott. Bianca Falcidieno della sede secondaria di Genova di CNR-IMATI, il cui interesse costante verso il mio progetto è stato fondamentale non solo per il tirocinio ma anche per i miei progetti futuri.

7. Riferimenti bibliografici

- [1] G. Pasqualotto, P. Zanuttigh, and G. M. Cortelazzo, "Combining color and shape descriptors for 3D model retrieval," *Signal Process. Image Commun.*, vol. 28, no. 6, pp. 608–623 (2013).
- [2] S. Biasotti, A. Cerri, M. Abdelrahman, M. Aono, A. Ben Hamza, M. El-Melegy, A. Farag, V. Garro, A. Giachetti, D. Giorgi, A. Godil, C. Li, Y.-J. Liu, H. Y. Martono, C. Sanada, A. Tatsuma, S. Velasco-Forero, and C.-X. Xu, "Retrieval and Classification on Textured 3D Models," in *3DOR 2014*, pp. 111–120 (2014).
- [3] S. Biasotti, A. Cerri, B. Falcidieno, and M. Spagnuolo, "3D Artifacts Similarity Based on the Concurrent Evaluation of Heterogeneous Properties," *J. Comput. Cult. Herit.*, vol. 8, no. 4, pp. 19:1–19:19, (2015).
- [4] T. Gevers, A. W. Smeulders, "Pictoseek: combining color and shape invariant features for image retrieval", *IEEE Trans. Image Process*, vol. 9(1), pp. 102–119, (2000)
- [5] R. Kimmel, R. Malladi, N. Sochen, "Images as embedded maps and minimal surfaces: movies, color, texture, and volumetric medical images", *Int. J. Comput. Vis.*, vol. 39(2), pp. 111–129 (2000)
- [6] H. Ling, D. Jacobs, "Deformation invariant image matching", *IEEE International Conference on Computer Vision (ICCV)*, vol. 2, pp. 1466–1473 (2005)
- [7] S. Biasotti, A. Cerri, A. Bronstein, M. Bronstein, "Quantifying 3D shape similarity using maps: recent trends, applications and perspectives", Lefebvre, S., Spagnuolo, M. (eds.) *EG 2014—STARs*, pp. 135–159 (2014)
- [8] J. Tangelder, R. Veltkamp, "A survey of content-based 3D shape retrieval methods", *Multimed. Tools Appl.*, vol. 39(3), pp. 441–471 (2008)
- [9] M. Suzuki, "A Web-based retrieval system for 3D polygonal models", *Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, vol. 4, pp. 2271–2276 (2001)
- [10] B. T. Phong, "Illumination for computer generated pictures", *Commun. ACM* vol. 18(6), pp. 311–317 (1975)
- [11] C. Ruiz, R. Cabredo, L. Monteverde, Z. Huang, "Combining shape and color for retrieval of 3D models", *Fifth International Joint Conference on INC, IMS and IDC (NCM'09)*, pp.1295–1300 (2009)
- [12] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, "Shape distributions", *ACM Trans. Graph.* vol. 21(4), pp. 807–832 (2002)
- [13] J. Starck, A. Hilton, "Correspondence labelling for wide-timeframe free-form surface matching", *IEEE International Conference on Computer Vision (ICCV)*, pp. 1–8 (2007)
- [14] T. Ojala, M. Pietikäinen, D. Harwood, "A comparative study of texture measures with

- classification based on featured distributions", *Pattern Recognit.* vol. 29(1), pp. 51–59 (1996)
- [15] D. G. Lowe, "Distinctive image features from scale-invariant key-points", *Int. J. Comput., Vis.* vol. 60(2), pp.91–110 (2004)
- [16] N. Dalal, B. Triggs, "Histograms of oriented gradients for human detection", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 886–893 (2005)
- [17] M. D. Fairchild, "Color Appearance Models", Wiley, New York (2005)
- [18] C. Wu, B. Clipp, X. Li, J. M. Frahm, M. Pollefeys, "3D model matching with viewpoint-invariant patches (VIP)", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8 (2008)
- [19] A. Zaharescu, E. Boyer, R. Horaud, "Keypoints and local descriptors of scalar functions on 2D manifolds", *Int. J. Comput. Vis.* vol. 100(1), pp. 78–98 (2012)
- [20] G. Cortelazzo, N. Orio, "Retrieval of colored 3D models", *Third International Symposium on 3D Data Processing, Visualization, and Transmission*, pp. 986–993 (2006)
- [21] A. Kanezaki, T. Harada, Y. Kuniyoshi, "Partial matching of real textured 3D objects using color cubic higher-order local auto-correlation features", *Vis. Comput.*, vol. 26(10), pp. 1269–1281 (2010)
- [22] Y. J. Liu, Y. F. Zheng, L. Lv, Y. M. Xuan, X. L. Fu, "3D model retrieval based on color+geometry signatures", *Vis.Comput.*, vol. 28(1), pp. 75-86 (2012)
- [23] F. Tombari, S. Salti, L. Di Stefano, "A combined texture-shape descriptor for enhanced 3D feature matching", *IEEE International Conference on Image Processing(ICIP)*, pp.809–812(2011)
- [24] N.J. Mitra, M. Pauly, M. Wand, D. Ceylan, "Symmetry in 3D geometry: extraction and applications", *Comput. Graph. Forum*, vol. 32(6), pp. 1–23 (2013)
- [25] S. Biasotti, A. Cerri, M. Abdelrahman, M. Aono, A. Ben Hamza, M. El-Melegy, A. Farag, V. Garro, A. Giachetti, D. Giorgi, A. Godil, C. Li, Y. J. Liu, H. Y. Martono, C. Sanada, A. Tatsuma, S. Velasco-Forero, C. X. Xu, "Retrieval and classification on textured 3D models", *Eurographics Workshop on 3D Object Retrieval (3DOR)*, pp. 111–120 (2014)
- [26] A. Giachetti, C. Lovato, "Radial symmetry detection and shape characterization with the multiscale area projection transform", *Comput. Graph. Forum*, vol.31(5), pp. 1669–1678 (2012)
- [27] G. Loy, A. Zelinsky, "Fast radial symmetry for detecting points of interest", *IEEE T. Pattern Anal.*, vol. 25(8), pp. 959–973 (2003)
- [28] A. Cerri, S. Biasotti, M. Abdelrahman, J. Angulo, K. Berger, L. Chevallier, M. El-Melegy, A. Farag, F. Lefebvre, A. Giachetti, H. Guermoud, Y. J. Liu, S. Velasco-Forero, J. Vigouroux, C. X. Xu, J. B. Zhang, "SHREC'13 track: retrieval on textured 3D models", *Eurographics Workshop on 3D Object Retrieval (3DOR)*, pp. 73–80 (2013)
- [29] D. Smeets, T. Fabry, J. Hermans, D. Vandermeulen, P. Suetens, "Isometric deformation modelling for object recognition", *Computer Analysis of Images and Patterns. Lecture Notes in Computer Science*, vol. 5702, pp. 757–765, Springer, Berlin (2009)
- [30] C. Li, A. Ben Hamza, "Symmetry discovery and retrieval of non-rigid 3D shapes using geodesic

skeleton paths”, *Multimed. Tools Appl.*, vol. 72(2), pp. 1027–1047 (2014)

- [31] Z. Lian, A. Godil, B. Bustos, M. Daoudi, J. Hermans, S. Kawa-mura, Y. Kurita, G. Lavoué, H. Van Nguyen, R. Ohbuchi, Y. Ohkita, Y. Ohishi, F. Porikli, M. Reuter, I. Sipiran, D. Smeets, P. Suetens, H. Tabia, D. Vandermeulen, “A comparison of methods for non-rigid 3D shape retrieval”, *Pattern Recognit.*, vol. 46(1), pp. 449–461 (2013)
- [32] C. Wang, M. M. Bronstein, A.M. Bronstein, N. Paragios, “Discrete minimum distortion correspondence problems for non-rigid shape matching”, *Proceedings of the Third International Conference on Scale Space and Variational Methods in Computer Vision (SSVM’11)*, pp. 580–591. Springer, Berlin (2012)
- [33] C. Li, “Spectral geometric methods for deformable 3D shape retrieval”, Master’s thesis, Concordia University, Montreal (2013)
- [34] M. Reuter, F. E. Wolter, N. Peinecke, “Laplace–Beltrami spectra as Shape-DNA of surfaces and solids”, *Comput. Aided Des.*, vol. 38(4), pp. 342–366 (2006)
- [35] K. Gebal, J. A. Bærentzen, H. Aanæs, R. Larsen, “Shape analysis using the auto diffusion function”, *Comput. Graph. Forum*, vol. 28(5), pp. 1405–1413 (2009)
- [36] J. Sun, M. Ovsjanikov, L. Guibas, “A concise and provably informative multi-scale signature based on heat diffusion”, *Comput. Graph. Forum*, vol. 28(5), pp. 1383–1392 (2009)
- [37] M. Aubry, U. Schlickewei, D. Cremers, “The wave kernel signature: a quantum mechanical approach to shape analysis”, *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp. 1626–1633 (2011)
- [38] R. M. Rustamov, “Laplace–Beltrami eigenfunctions for deformation invariant shape representation”, *Proceedings of the Fifth Eurographics Symposium on Geometry Processing (SGP’07)*, pp. 225–233. Eurographics Association, Aire-la-Ville (2007)
- [39] C. Li, A. Ben Hamza, “A multi resolution descriptor for deformable 3D shape retrieval”, *Vis. Comput.*, Vol. 29, pp. 513–524 (2013)
- [40] A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, R. Kimmel, “Photometric heat kernel signatures”, *Proceedings of the Third International Conference on Scale Space and Variational Methods in Computer Vision (SSVM’11)*, pp. 616–627. Springer, Berlin (2012)
- [41] A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, D. Raviv, R. Kim-mel, “Affine invariant photometric heat kernel signatures”, *Eurographics Workshop on 3D Object Retrieval (3DOR)*, pp. 39– 46 (2012)
- [42] A. Kovnatsky, D. Raviv, M. Bronstein, A. M. Bronstein, R. Kim-mel, “ Geometric and photometric data fusion in non-rigid shape analysis”, *Numer. Math. Theor. Methods Appl.*, vol. 6(1), pp. 199– 222 (2013)
- [43] S. Biasotti, A. Cerri, D. Giorgi, M. Spagnuolo, “PHOG: pho-tometric and geometric functions for textured shape retrieval”, *Comput. Graph. Forum* vol. 32(5), pp. 13–22 (2013)
- [44] H. Edelsbrunner, J. Harer, “Persistent homology a survey”, *Surveys on Discrete and Computational Geometry, Contemp. Math.*, vol. 453, pp. 257–282. Am. Math. Soc., Providence (2008)
- [45] P. Frosini, C. Landi, “Size theory as a topological tool for computer vision”, *Pattern Recognit.*

Image Anal., vol. 9(4), pp. 596–603 (1999)

- [46] S. Biasotti, D. Giorgi, M. Spagnuolo, B. Falcidieno, “Reebgraphs for shape analysis and applications”, *Theor. Comput. Sci.*, vol. 392(1–3), pp. 5–22 (2008)
- [47] G. Reeb, “Sur les points singuliers d’une forme de Pfaff complètement intégrable ou d’une fonction numérique”, *Comptes Rendus Hebdomadaires des Séances de l’Académie des Sciences*, vol. 222, pp. 847–849 (1946)
- [48] S. Biasotti, D. Giorgi, M. Spagnuolo, B. Falcidieno, “Size functions for comparing 3D models”, *Pattern Recogn.*, vol. 41(9), pp. 2855–2873 (2008)
- [49] D. Cohen-Steiner, H. Edelsbrunner, J. Harer, “Stability of persistence diagrams”, *Discret. Comput. Geom.*, vol.37(1), pp. 103–120 (2007)
- [50] C. Li, M. Ovsjanikov, F. Chazal, “Persistence-based structural recognition”, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1995–2002 (2014)
- [51] A. Cerri, C. Landi, “The persistence space in multidimensional persistent homology”, Gonzalez-Diaz, R., Jimenez, M.J., Medrano, B. (eds.) *Discrete Geometry for Computer Imagery. Lecture Notes in Computer Science*, vol. 7749, pp. 180–191. Springer, Berlin (2013)
- [52] S. Biasotti, A. Cerri, M. Abdelrahman, M. Aono, A. Ben Hamza, V. Garro, A. Giachetti, D. Giorgi, A. Godil, C. Li, C. Sanada, M. Spagnuolo, A. Tsumura, S. Velasco-Forero, “Retrieval and classification methods for textured 3D models: a comparative study”, in *Vis. Comp.* 2016, 2016, vol. 32, pp. 217–241
- [53] E. Albu, E. D. Kocalar, A. A. Khokhar, "Quantized CIE Lab* Space and Encoded Spatial Structure for Scalable Indexing of Large Color Image Archives "
- [54] Van Rijsbergen, Cornelis Joost "Keith" (1979); *Information Retrieval*, London, GB; Boston, MA: Butterworth, 2nd Edition
- [55] R. R. Coifman and S. Lafon, “Diffusion maps,” *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 5–30, 2006.
- [56] O. Sidi, O. van Kaick, Y. Kleiman, H. Zhang, and D. Cohen-Or, “Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering,” *ACM Trans. Graph.*, vol. 30, no. 6, pp. 126:1–126:10, 2011.
- [57] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *KDD*, 1996, pp. 226–231.
- [58] S. Biasotti, M. Spagnuolo, and B. Falcidieno, “Grouping real functions defined on 3D surfaces,” *Comput. Graph.*, vol. 37, no. 6, pp. 608–619, 2013.

Recent titles from the IMATI-REPORT Series:

16-01: *Optimal strategies for a time-dependent harvesting problem*, G.M. Coclite, M. Garavello, L.V. Spinolo, February 2016.

16-02: *A new design for the implementation of isogeometric analysis in Octave and Matlab: GeoPDEs 3.0*, R. Vázquez, April 2016.

16-03: *Defect detection in nanostructures*, D. Carrera, F. Manganini, G. Boracchi, E. Lanzarone, April 2016.

16-04: *A study of the state of the art of process planning for additive manufacturing*, M. Livesu, M. Attene, M. Spagnuolo, B. Falcidieno, May 2016.

16-05: *Calcolo di descrittori ibridi geometria-colore per l'analisi di similarità di forme 3D*, A. Raffo, S. Biasotti, June 2016.