Combining color and shape descriptors for 3D model retrieval

GIULIANO PASQUALOTTO, PIETRO ZANUTTIGH and GUIDO M. CORTELAZZO

Dept. of Information Engineering, University of Padova

Abstract

Nowadays many three dimensional models feature color information together with the shape description. However current content-based retrieval schemes for 3D models are based on shape information only and ignore color clues. The significance of shape versus color clues for 3D model retrieval is instead a fundamental issue still almost unexplored at this time. A possible approach is to extend shape-based 3D model retrieval methods of proven effectiveness in order to include color. This work follows such rationale and introduces an extended version of the spin-image descriptor that can account also for color data. The comparison of color descriptors is performed using a novel scheme that allows to recognize as similar also objects with different colors but distributed in the same way over the shape. Shape and color similarity are finally combined together by an algorithm based on fuzzy logic. Experimental results show how the joint use of color and shape data allows to obtain better results than each of the two types of information alone. Comparisons with state-of-the-art content-based retrieval methods for 3D models also show how the proposed scheme outperforms standard solutions on object classes with meaningful color information.

Keywords: Content-based retrieval, Spin-Images, Color, Fuzzy logic, 3D models

1. Introduction

Nowadays more and more 3D models are becoming available on the web and large 3D databases, e.g. Google's 3D warehouse [1], which requires efficient searching techniques tailored to this type of data. Many different schemes have been proposed for content-based retrieval of 3D data based on different feature descriptors representing both global and local properties of the objects' shape. However most current content-based retrieval schemes for 3D models exploit only shape information while other attributes, especially the color information, are not considered for retrieval purposes. In fact most of the currently available 3D models have also color information associated to them which is a very relevant clue in order to distinguish between different objects. A possible approach in order to exploit color information is to extend standard descriptors used to represent the shape of the objects in order to encompass also the color data and in particular its distribution over the shape.

This paper builds over this rationale and presents a novel content-based retrieval method for 3D models that exploits both shape and color information. Following the idea firstly introduced in [2], an extended version of the spin-image descriptor [3] that can account also for color information is presented. This is not the only point of interest of this work, since combining shape and color clues for 3D model retrieval introduces other original elements. Among them there is a new way of computing the similarity between the color spin-image descriptors that allows to recognize as similar also object with different colors but distributed in the same way over the shape, differently from [2] where this type of objects were considered different. This paper introduces also a global index combining shape and color similarity based on fuzzy sets theory. The experimental results confirm the effectiveness of using color information together with shape information by the proposed method. Indeed the proposed technique when compared with state-of-the-art content-based retrieval methods for 3D models outperforms them on object classes exhibiting meaningful color data.

After reviewing the related work in Section 2, the rest of the paper is organized as depicted in Fig. 1. Section 3 introduces the proposed shape descriptor. Section 4 builds the color descriptor on top of the shape one. Section 5 presents the proposed similarity measures for shape and color information and an efficient way to combine the two measures. Then the experimental results are shown in Section 6 and finally Section 7 draws the conclusions.

2. Related work

Although content-based retrieval of 3D models is a quite recent research field, many different methods have been proposed, some of them will be

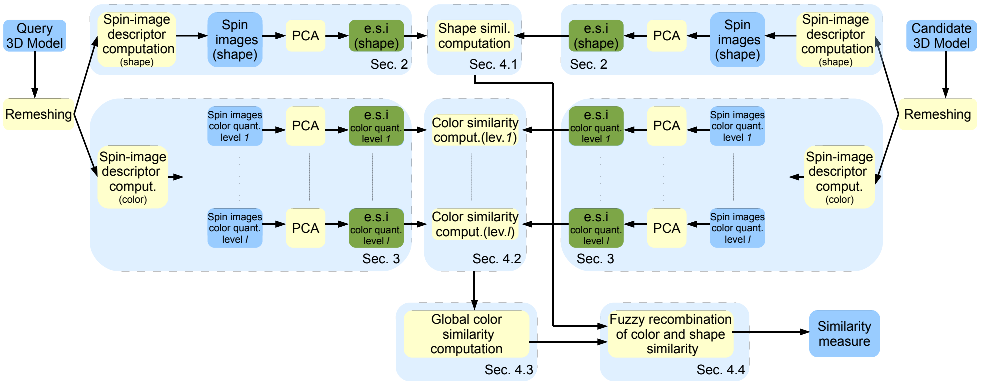

Query 3D Model
Remeshing
Spin-image descriptor computation (shape)
Spin images (shape)
PCA
e.s.i (shape)
Sec. 2
Shape simil. computation
Sec. 4.1
Spin-image descriptor comput. (color)
Spin images color quant. level 1
PCA
e.s.i color quant. level 1
Spin images color quant. level l
PCA
e.s.i color quant. level l
Sec. 3
Color similarity comput.(lev.1)
Color similarity comput.(lev.l)
Sec. 4.2
e.s.i (shape)
PCA
Spin images (shape)
Spin-image descriptor computation (shape)
Sec. 2
Candidate 3D Model
Remeshing
e.s.i color quant. level 1
PCA
Spin images color quant. level 1
e.s.i color quant. level l
PCA
Spin images color quant. level l
Spin-image descriptor comput. (color)
Sec. 3
Global color similarity computation
Sec. 4.3
Fuzzy recombination of color and shape similarity
Sec. 4.4
Similarity measure

Figure 1: Architecture of the proposed content-based 3D model retrieval system.

briefly recalled next. Extended reviews of the literature in this field can be found in [4], [5], [6] and [7] while a very recent benchmark of the best performing techniques is presented in [8]. Content-based retrieval methods usually encompass first the extraction of a set of feature descriptors from each object and then their comparison for searching purposes. In the case of 3D models these features need to be highly descriptive as well as invariant with respect to many transformations such as changes in the object's pose, scale or mesh resolution.

A first group of methods uses global features characterizing the whole shape of the objects. In [9] a set of different global descriptors was proposed, including cords-based, moments-based, and also wavelets-based features. Other global descriptors use voxel-based spherical harmonics [10] or spherical wavelets [11]. There are also solutions that use statistical models based on the distribution of the vertices in the 3D space: Osada et Al. [12] introduced a set of shape functions measuring distances, angles, areas and volumes between random surface vertices. Such distributions are then compared in order to evaluate object similarities. This method was extended in [13] by computing the distances on a regular voxel grid. Gao et al. [14] introduced the spatial structure circular descriptor, that allows to capture the spatial structure of the 3D model into a set of images that can then be compared for retrieval purposes. Another global method, the *Relativistic Impact Descriptor* [15] associates to each 3D object a curved space modified by its mass from which two invariant descriptors are extracted.

An alternative class of approaches is based on the use of local features.

These methods exploit descriptors representing the surface in the proximity of a vertex, like the *bag of features* based approach of [16], the *3D Shape Spectrum Descriptor* used within the MPEG-7 framework or the spin-image descriptor, firstly introduced in [3] for object matching and then used for 3D retrieval in the schemes of [17] and [18]. There are also methods that combine both local and global descriptors, e.g., in [19] the global 3D shape is represented by the spatial configuration of a set of specific local shapes.

Finally view-based methods represent 3D objects by a set of views taken from different viewpoints and then exploit content-based image retrieval techniques to perform the comparison. In [20] 5 different groups of views are extracted from the model and then a probabilistic approach is used to find the models that maximize the a posterior probability given the query model. The method of [21] instead extracts 2D rotation-invariant shape descriptors from a set of views and combines this information into a global shape similarity measure. Chen et al. [22] compute the light fields of the 3D objects and extract the descriptors from their silhouettes. A similar scheme based on spherical correlation [23] offers better performances and overcomes some limitations of the approach of Chen et al. The views can be standard images but it is also possible to describe the three dimensional structure of the object through image representations. Example of this approach are depth maps or of the above mentioned spin-image descriptor which represents 3D point displacements by way of images. Depth maps obtained from the rendering of the 3D object have been used in several retrieval approaches (e.g. [24]). Multiple descriptors can also be combined together, e.g., the work of Daras et Al. [25] exploits together several different descriptor. The paper investigates how to select the optimal similarity measure for descriptor comparison. It also presents a method to combine together results from different descriptors by optimizing the relative weights through particle swarm optimization (PSO).

All the above methods exploit only the objects shape and ignore possible appearance attributes linked to the vertices of the objects, like color, material, or texture. While many methods have been developed for shape-based retrieval of 3D objects and a huge amount of work has been devoted to content-based retrieval of color images, the joint use of shape and color data for retrieval purposes is still an almost unexplored field. This intriguing possibility has been considered in [26] where color information represented by the Phong's model parameters is used to assist in the retrieval process. Color and shape similarity are used together also in [27]. Here a shape similarity measure based on the method of [12] is combined with color similarity computed through the comparison of color distribution histograms. However in this method the color is considered as a general property and its distribution over the shape is not considered. Furthermore the balancing of the two clues is done by using a manually tuned weighting factor. Another scheme combining both color and geometry information has been recently presented in [28]. In this work salient feature points are extracted on the basis of both color and geometry information. Then a shape signature is computed on these points and used for the comparison. The use of color information in this work however remains limited to the selection of feature points, while the descriptors are computed only on the basis of geometrical distances. Finally in [2] a color extension of the spin-image descriptor is proposed for colored 3D objects retrieval. This approach is basically an adaption of the colored 3D data registration scheme of [29] to the retrieval field. There are substantial differences between the method of [2] and the method proposed in this manuscript which also exploits spin-image descriptors for 3D retrieval. Firstly in [2] pure shape descriptors are not considered together with color information as in the proposed method. Furthermore the color descriptor of [2] considers only luminance values without the chromatic components. Finally and more importantly in [2], as in most of the other works exploiting color information, similar objects of different color are considered different, while the proposed approach considers the color distribution over the shape rather than simply the color values by themselves. In this way the proposed method is able to recognize as similar objects of similar shape and color distribution even if they have different colors.

3. Construction of the 3D shape descriptor

As previously said the proposed method retrieves 3D objects on the basis of both color and shape information. Fig. 1 shows its global architecture. Following a quite common approach there are two main stages: the construction of a set of descriptors for each object and the comparison of the descriptors in order to compute the similarity between two objects. For each model that needs to be compared two sets of descriptors are built (shown in green in the figure): one including shape information only and another one also encompassing color data. Then, in order to compare two 3D models, their shape and color descriptors are compared independently with the methods described in Section 5 leading to a shape and a color similarity measure respectively. Finally a recombination stage based on fuzzy logic takes as input the two measures and computes a final joint similarity score.

Since the color-based descriptor is built on top of the shape-based one, it is appropriate to introduce first the shape only descriptor in this section. The proposed shape descriptor construction features three main steps: an initial pre-processing stage; the construction of the descriptors (an approach based on spin-images [30] has been used); and a further processing of such descriptors based on PCA. Two operations are performed on each model in the initial pre-processing stage: first a remeshing and then a rescaling. The remeshing is performed in order to obtain models with a similar number of vertices and a rather uniform distribution of the vertices. Indeed the spin-image descriptors [30], that will be used in the following stage, measure the density of vertices. Therefore without such a pre-processing step, a 3D model with a large number of vertices in one region and large flat areas represented by a few large triangles can be considered different from a similar 3D model with the flat region densely sampled. This can be avoided by a remeshing stage, which limits the length of the mesh edges within a given interval. Color information is also taken into account: in the edge-split operation the color of each new vertex is given by the mean of the color values associated to the edge vertices, while in the edge-collapse operation the color of the new vertex is defined as the linear combination of the colors of adjacent vertices with coefficients proportional to their Euclidean distance from the new vertex. The adopted remeshing algorithm is an improvement of the approach used in [31] that iteratively applies the edge-collapse and edge-split operations to the edges exceeding the desired interval. The number of vertices in the remeshed models is a parameter of the algorithm that allows to trade-off between the computation time and the accuracy of the spin-image descriptors. The results get better by increasing the number of vertices even if the computation time also increases. Note also that after reaching a certain point no further improvement is obtained by increasing the number of vertices. For the scope of this work we remeshed the models in order to compute descriptors over objects with around 5000 vertices each. This size leads to performances quite close to the optimal values while still allowing for a fast computation of the descriptors. After remeshing, each 3D model is rescaled in order to have a bounding box with unit value diagonal. This allows to compare models of different scales or with different coordinate systems.

The second step is the computation of the shape descriptor, which is based on the spin-images descriptor originally introduced by [30] for 3D registration. The concept of spin-image is here quickly recalled for presentation's clarity; the reader is referred to [3] and [30] for the details. An oriented point at a surface mesh vertex is defined as the pair formed by the 3D vertex coordinates $\mathbf{p}$ and the surface normal $\mathbf{n}$ at the vertex. For any point $\mathbf{x}$ in the 3D space, define a coordinate pair $(\alpha;\beta)$ with respect to the reference system $(\mathbf{p};\ \mathbf{n})$ associated to the oriented point in the following way: radial coordinate α is the distance from $\mathbf{x}$ to the line corresponding to the surface normal $\mathbf{n}$, elevation coordinate β is the signed perpendicular distance from $\mathbf{x}$ to the plane through $\mathbf{p}$ and perpendicular to $\mathbf{n}$. The record of the $(\alpha;\beta)$ coordinates of all the points of the 3D mesh, called "spin-map", is formally defined as:

$$SM(\mathbf{x})\leftarrow(\alpha,\beta)=(\sqrt{\|\mathbf{x}-\mathbf{p}\|^2-(\mathbf{n}\cdot(\mathbf{x}-\mathbf{p}))^2},\mathbf{n}\cdot(\mathbf{x}-\mathbf{p})) \qquad (1)$$

A spin-image is a spatially discretized version of a spin-map, where the gray value at each location is associated to the count of points of the spin-map falling in each discrete cell (*bin*). Since we are interested in the global shape of the object and not in simply matching some surface regions as in [30], the support of the spin-images is chosen large enough to encompass the whole object. The number of quantization levels in the α and β dimensions will be denoted by N_a and N_b respectively. Fig. 2a shows a 3D view of an object, Fig. 2b its spin-map with respect to the reference point marked on Fig. 2a and Fig. 2c the corresponding spin-image. As Fig. 2 exemplifies, spin-images depend only on the intrinsic surface characteristics and not on the surface's spatial position and orientation. In other words the spin-image associated with a vertex point is invariant with respect to rigid rototranslations. Therefore a collection of images can be associated to a 3D mesh by way of spin-images, as every surface point can generate a spin-image. The N_a and N_b values determine the accuracy of the descriptor. By increasing these number a finer representation of the object is obtained. However a too fine quantization can lead to empty bins or bins with just a very few vertices, besides increasing also the descriptor size and the computation time. For the retrieval purposes we are just interested in the global object appearance and $N_a = N_b = 10$ proves to be a good trade-off between the descriptor accuracy and the memory requirements. However the optimal value of N_a and N_b depends on the number of vertices used in the remeshing stage, a fine quantization with a coarse remeshing is not very meaningful.

Let us consider a 3D model with v vertices. A spin-image can be built for each vertex and by reading it in row order it can be mapped into vector $\mathbf{s_i}$ of

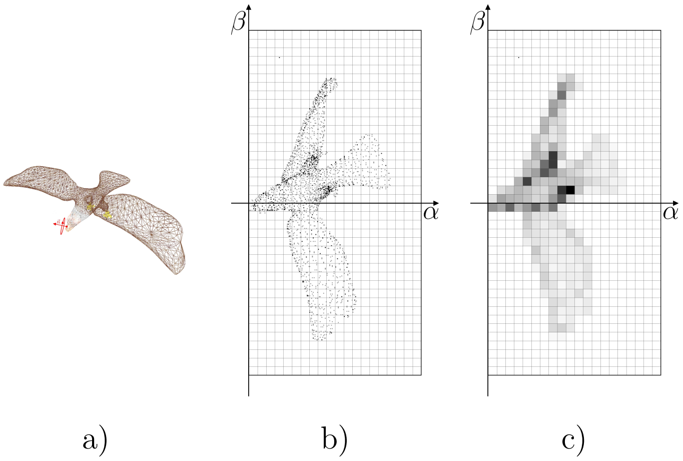


Figure 2: a) Sample 3D model with an highlighted vertex; b) Spin-map associated to the highlighted vertex; c) Corresponding spin image.

size $d = N_a \times N_b$. Hence, spin image computation associates to the 3D model a $d \times v$ matrix $S = [\mathbf{s_1}, \ldots, \mathbf{s_v}]$ with a column for each vertex. PCA can then be applied to S since the interest is for the global properties of the model. Let us call eigen spin-images the resulting eigenvectors. Thus a set of v eigen spin-images E_i can be associated to each 3D model. To reduce descriptor size and computation times each 3D object can be associated to the set of the $c_{sh} < v$ eigenvectors corresponding to the greatest eigenvalues. Parameter c_{sh}, called the *cut-off* of the shape descriptor, is associated to the number of eigenvectors ensuring that the reconstruction error is below a pre-defined threshold e_{ms} [30]. Threshold e_{ms} is a fundamental parameter for spin-image based approaches since it allows to trade-off between descriptors size and the computational requirements on one side and representation accuracy on the other. In the experimental results of this paper we set $e_{ms} = 0,05$ (the reconstruction error is measured on the correlation between the original and reconstructed descriptors, see [30] for more details) in order to achieve nearly optimal retrieval performances and avoid too large memory and computation time requirements. However if a faster computation time is required, e.g., for searching on very large databases, different values of e_{ms} can be selected in order to find the preferred trade-off between computation time, descriptor size and accuracy of the retrieval. In summary the shape descriptor stored for each 3D model $\mathcal{M}$ is given by

$$\begin{aligned} E &= [\mathbf{e}_1, \ldots, \mathbf{e}_{c_{sh}}] \\ \mathbf{V}_{sh} &= [\lambda_1, \ldots, \lambda_{c_{sh}}] \end{aligned} \tag{2}$$

where E is the matrix of the eigen spin-images and $\mathbf{V}_{sh}$ the corresponding eigenvalues vector.

4. Construction of the 3D color descriptor

In the following we will assume that per-vertex color information is available (i.e., each vertex has a color value associated to it). If the 3D object color is represented by a texture map the color is simply mapped from the texture to the object vertices. This section shows how to extend the spin-images shape descriptor to include also color information. A first extension of the spin-images descriptor to include color information was proposed in [29]. However that approach was developed for the registration of 3D views and it is not suited to 3D object retrieval. An adaption of the ideas of [29] to 3D retrieval of colored objects has then been proposed in [2], but the method of [2] has severe limitations confining its application to the recognition of objects of the same color. The basic idea behind the approach proposed in this work is to quantize the color space into l levels and to build l eigen spin-images sets, one for each quantized region of the color space. The first step is the spin-images generation. For each mesh vertex, $(l + 1)$ spin-images are computed, one spin-image for each of the l color quantization levels plus one for the shape descriptor. The shape descriptor is built as explained in Section 3 while the l color spin-images are built as follows: given a reference vertex, each of the remaining vertices contributes a single unitary bin increment to the spin-image corresponding to its color quantization level. In other words each spin-image corresponding to a color quantization level accumulates only the vertices belonging to that color interval. Note also that the spin-image associated to the shape corresponds to the sum of the l color spin-images.

It is worth mentioning that the quantization strategy is a crucial aspect of the proposed 3D model retrieval system, since it determines the descriptiveness of each set of spin-images. On one hand, empty descriptor sets or sets with just a few vertices are not only useless for comparison purposes, but also cause a waste of computation time and memory. On the other hand, a too coarse quantization of the color space reduces the amount of information that can be extracted from color data. After testing several color quantization approaches we decided to adopt a modified version of the method proposed in [32]. Although this scheme is simple, it appears well suited for the coarse quantization necessary in our case. The adopted color quantization works in the following way: first of all, the colors of the vertices are converted to the CIELAB color space (a uniform color space ensures that Euclidean color distances are meaningful). The CIELAB space is then divided into 6 different regions as shown in Table 1. The first one includes all the dark colors (the ones with luminance value $L < 10$), while the sixth one has all the brightest colors ($L > 98$). The central luminance region is instead divided into four regions on the basis of the hue angle $h_{ab} = \arctan(b/a)$ as proposed in [32]. The subdivision is similar to the one proposed in [32] but we found beneficial to divide the region containing the very dark and very bright colors into two regions, one for the dark and one for the bright colors in order to better distinguish dark and bright regions in the objects. Some examples of sample colors falling in each quantization interval are shown in Fig. 3.

Quantization level	luminance (L)	hue angle (h_{ab})
1	$L < 10$	$\forall$
2	$10 < L < 98$	$-0,35 < \theta < 1,4$
3	$10 < L < 98$	$1,4 < \theta < 1,92$
4	$10 < L < 98$	$1,92 < \theta < 3,42$
5	$10 < L < 98$	$3,42 < \theta < 5,93$
6	$L > 98$	$\forall$

Table 1: Color quantization intervals

Once the $l + 1$ spin-images sets are computed, it is necessary to apply $(l + 1)$ PCAs, one for each spin-images set in order to obtain $(l + 1)$ sets of eigen spin-images. Each set of eigen spin-images has its own number of eigenvectors, depending on the cut-off of that color quantization level, evaluated similarly to the shape case treated in Section 3. It is important to note that the cut-off can be different for each color quantization level of each 3D object. The number of eigen-vectors used for the i_th set of eigen spin-images is called the *cut-off* of the i_th eigen spin-images level and it is denoted by symbol c_i. Hence, the final shape and color descriptor of the generic 3D model $\mathcal{M}$ features the $(l + 1)$ sets of eigen spin-images together with the associated $(l + 1)$ sets of eigenvalues and it can be represented by pair (C, V_{col}).

$$C = \{\overbrace{[\mathbf{e}_{11}, \dots, \mathbf{e}_{1c_1}]}^{E_1}, \overbrace{[\mathbf{e}_{21}, \dots, \mathbf{e}_{2c_2}]}^{E_2}, \dots, \overbrace{[\mathbf{e}_{l1}, \dots, \mathbf{e}_{lc_l}]}^{E_l}\}$$

$$V_{col} = \{\overbrace{[\lambda_{11}, \dots, \lambda_{1c_1}]}^{\mathbf{V}_1}, \overbrace{[\lambda_{21}, \dots, \lambda_{2c_2}]}^{\mathbf{V}_2}, \dots, \overbrace{[\lambda_{l1}, \dots, \lambda_{lc_l}]}^{\mathbf{V}_l}\} \tag{3}$$

where matrix $C = [E_1, \dots, E_l]$ has the l matrices containing the eigen spin-images of each quantization level. Therefore $E_i \in \Re^{d \times c_i}$, $1 \leq i \leq l$, where $d = N_a \times N_b$ is the total number of bins of a single spin-image and c_i is the cut-off value of the i_th quantization level. V_{col} has instead l vectors made by the eigenvalues associated to each eigen spin-images set. Let us finally point out that what we call "color" descriptor in fact represents the shape of the regions associated to a specific color quantization level, therefore in this respect it can be regarded as a combined shape and color descriptor. Thus, it is possible to directly use it for retrieval purposes without any shape descriptor. However we found that it is more efficient to combine it with a shape descriptor in order to handle objects with similar shape but a completely different color distribution or objects with a complex texture pattern.

5. Similarity measure

The final step is the computation of the similarity measure on the basis of the descriptors introduced in the previous sections. Let us consider a standard 3D model retrieval situation where $\mathcal{M}_q$ is a given *query* model and $\mathcal{C} = \{\mathcal{M}_n; n = 1, 2, ..., N\}$ is the set of all *candidate* 3D models. For retrieval purposes it is firstly necessary to compute a similarity measure between $\mathcal{M}_q$ and each $\mathcal{M}_n, n = 1, 2, ..., N$ using the descriptors introduced in the previous sections. To this purpose, in the proposed approach, the shape and color descriptors are first independently compared, then the two distance measures are combined together using fuzzy logic as described in Subsection 5.4.

5.1. Shape comparison

For the sake of clarity let us start from the shape only case: according to (2) each shape descriptor consists of a matrix $E = [\mathbf{e}_1, \dots, \mathbf{e}_{c_{sh}}]$ of eigen spin-images, with $\mathbf{e}_i \in \Re^n$, and by the vector $\mathbf{V}_{sh} = [\lambda_1, \dots, \lambda_{c_{sh}}]$ of their eigenvalues, where n is the total number of bins (of a single spin-image) and c_{sh} is the shape cut-off value. For the comparison between the query model

$\mathcal{M}_q(E_q,\ \mathbf{V}_{sh,q})$ and each candidate model $\mathcal{M}_n(E_n,\ \mathbf{V}_{sh,n})$, shape similarity is given by the expression:

$$Sim_{sh}(\mathcal{M}_q,\ \mathcal{M}_n) = \frac{\sum_{i=1}^{c_{\text{off}}} w(\lambda_{i,q})\ \|\rho(\mathbf{e}_{i,q},\ \mathbf{e}_{i,n})\|}{\sum_{i=1}^{c_{\text{off}}} w(\lambda_{i,q})} \tag{4}$$

where $c_{\text{off}} = \min(c_{sh,q}, c_{sh,n})$ is the minimum between the two cut-off values $c_{sh,q}$ and $c_{sh,n}$ of the query and candidate object respectively. The minimum has been chosen in order to compare eigenvectors that are significant (and has been stored) for both the compared models. Symbol ρ represents the statistical correlation function, while $w : V_s \rightarrow [0\ , 1]$ is a weighting function used to take into account the different descriptiveness of the eigen spin-images. In the proposed approach $w(\lambda_{i,q})$ is defined as:

$$w(\lambda_{i,q}) = \lambda_{i,q}^{\theta}, \quad 0 < \theta \leq 1. \tag{5}$$

Definition (5) is motivated by the idea of using the eigenvalues as linear combination coefficients for the correlations, since each eigenvalue reflects how representative the associated eigenvector is. More precisely each eigenvalue is proportional to the fraction of the data variance that is associated with the corresponding eigenvector and by weighting the eigenvectors with the corresponding eigenvalues we give more relevance to the eigenvectors that correspond to the larger variance of the data, i.e. to the eigenvectors corresponding to the most important structures of the 3D models. It was experimentally found that weights $\lambda_{i,q}^{\theta}$ with $\theta < 1$ are preferable to the direct use of $\lambda_{i,q}$ as weights (i.e., $\theta = 1$) in order to balance the very fast decay, which is typical of the eigenvalues. Indeed the usage of $\theta = 1$ was found prone to loose the descriptiveness of some small details, corresponding to small fractions of the object surface nevertheless associated to very descriptive elements for some classes of models (e.g., the legs for the ant class of the *Princeton Shape Benchmark* [33]).

Finally note how, since the weights $w(\lambda_{i,q})$ depend on the eigenvalues of the query model only, the proposed similarity measure is not perfectly symmetric, i.e. given two models $\mathcal{M}_a$ and $\mathcal{M}_b$, in general $Sim_{sh}(\mathcal{M}_a,\ \mathcal{M}_b) \neq Sim_{sh}(\mathcal{M}_b,\ \mathcal{M}_a)$. The rationale behind the adopted similarity measure is the following: in typical retrieval applications the query model is compared against all the other models in order to produce a results list ranking all the models, from the most to the least similar with respect to the query model. The choice of the eigenvectors of the query model as weights ensures an unbiased and *fair* comparisons, since for each query the correlations of all the other models are weighted by the same set of weights. Furthermore, since similar objects of the same category have also similar eigenvalues, the models in the first positions of the query results (which usually are the interesting ones in retrieval applications) $Sim_{sh}(\mathcal{M}_a,\ \mathcal{M}_b) \sim Sim_{sh}(\mathcal{M}_b,\ \mathcal{M}_a)$, i.e. the similarity measure has very similar values. In Section 6 it will be shown also from the experimental point of view that this issue is not critical and that the outcomes of different queries are consistent.

5.2. Accounting for spatial color distribution

The target of the color comparison procedure is to evaluate the similarity of the different color regions of the objects in order to recognize objects with similar color structure over their shape. The idea behind the proposed approach is to compare each eigen spin-images set corresponding to a given color quantization level with the eigen spin-images sets of all the other color quantization levels. The rationale for this approach is that the eigen spin-images of regions with different colors but similar color distribution on their shape (i.e., corresponding to objects with different colors but similarly distributed on the shape) are highly correlated. To clarify this aspect consider two objects of the same class but with different colors, e.g. the two people wearing clothes of different colors and in a slightly different pose shown in Fig. 3. They will probably have some regions with the same color, e.g., the face, but others of similar shape and extension but with different colors (e.g., their trousers). By comparing each region with all the others it is possible, for example, to match the black trousers of a man with the blue ones of another person. This fact is exemplified by the highlighted eigen spin-images of Fig. 3 where the left-most eigen spin-image of the top row (dark colors) is very similar to the fifth eigen spin-image from the left of the second row (bluish colors). Note how also the other eigen spin-images corresponding to the various color regions and parts of the body are quite similar even if not so similar due to the different pose of the two beings and of the different captured areas.

In order to compare the color descriptor of the query model $\mathcal{M}_q(C_q, V_{col,q})$ with the one of a candidate model $\mathcal{M}_n(C_n, V_{col,n})$, according to the notation introduced in (3) the proposed comparison method considers the maximum similarity between the i_th eigen spin-images matrix $E_{i,q}$ of the query descriptor ($E_{i,q} \in \Re^{n \times c_{i,q}},\ 1 \leq i \leq l$) and all the eigen spin-images matrices $E_{j,n}, j = 1, 2, .., l$ of the candidate model. Hence the color similarity for the

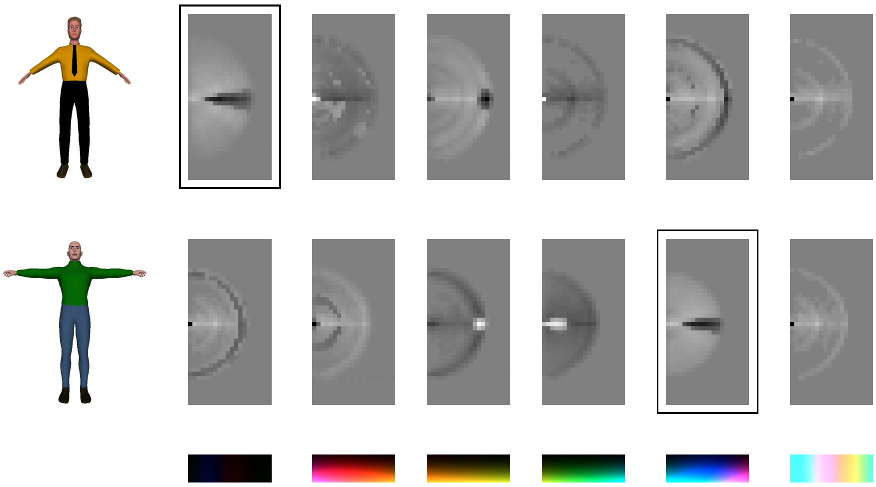


Figure 3: Example of color eigen spin-images for each color quantization interval of two sample models belonging to the *human* class. The first two rows show the two models together with the corresponding color eigen spin-images, while the third row shows some sample colors from the color quantization interval corresponding to each eigen spin-image.

i_th color quantization level is defined as:

$$Sim^{i}_{col}(\mathcal{M}_q,\mathcal{M}_n) = \max_{1\leq j\leq l}\left\{\frac{\sum_{k=1}^{c_{ij}} w(\lambda_{ik,q})\,\|\rho(\mathbf{e}_{ik,q},\mathbf{e}_{jk,n})\|}{\sum_{k=1}^{c_{ij}}\ w(\lambda_{ik,q})}\right\} \tag{6}$$

where

$$c_{ij} = \min\{c_{i,q},\ c_{j,n}\} \tag{7}$$

is the minimum between $c_{i,q}$ and $c_{j,n}$. Hence, given an eigen spin-images set $E_{i,q}$ (i.e., a color quantization level) of the query descriptor, the similarity comparison evaluates all the pairs of eigen spin-images sets $(E_{i,q},\ E_{j,n}), j = 1,..,l$, where $E_{j,n}$ is the eigen spin-images set of the j_*th* color quantization level of the candidate descriptor. The similarity value of the i_th color level corresponds to the maximum similarity between the pairs made by the i_th e.s.i set of the query object and each one of the eigen spin-images sets of the candidate object. This procedure allows to compute l color similarity values $Sim^{i}_{col}(\mathcal{M}_q,\ \mathcal{M}_n), i = 1,..,l$ for each query model which correspond to the different color regions.

5.3. Weighting color quantization levels

The l color similarity values correspond to different regions possibly with different size and relevance for the query model. It is therefore necessary to consider a different weight for each color quantization level, accounting for its descriptiveness. Following the same approach used to weight the eigenvectors in the shape descriptor, once fixed the number of eigenvectors (i.e., after computing the minimum cut-off) for each color quantization level, the descriptiveness considered in the proposed method is the sum of the associated eigenvalues. As previously underlined the rationale behind this choice is that by weighting the eigenvectors with the corresponding eigenvalues we give more relevance to the eigenvectors that correspond to the larger variance of the data. By adding the first c (ordered) eigenvalues[1], indeed, it is possible to determine the relevance of the subset given by the first c most significant eigenvectors. In this manner high relevance, evaluated against the entire set of eigenvectors, indicates high descriptiveness of that subset. This comes from the assumptions that the most significant eigenvectors are also the most descriptive ones and that their eigenvalues are related to such descriptiveness. Such assumptions are experimentally confirmed. From the above, the weighting coefficients of the different color quantization levels are defined as:

$$w_{l_i} = \sum_{k=1}^{c_{i,q}} \lambda_{ik,q}^{\theta}, \quad 0 < \theta \leq 1. \tag{8}$$

where $\lambda_{ik,q}$ indicates the k_th (ordered) eigenvalue of the query model descriptor relative to color quantization level i, and $c_{i,q}$ is the cut-off of the i_th color quantization level of the query object. In this manner the weights are only determined by the eigenvalues of the query object; this ensures that the same weighting is used for the comparison with all the candidate models and that the comparison is "fair" even though the similarity measure is not perfectly symmetric as noted in Subsection 5.1.

After computing the $Sim_{col}^{i}(\mathcal{M}_q,\ \mathcal{M}_n)$ value of each color quantization level, the global color similarity is computed as the sum of the similarity values weighted by the corresponding w_{l_i} coefficients from Equation (8) suitably

[1] It is important to notice that the computation considers normalized eigenvalues, i.e. eigenvalues which add up to one over each eigen spin-images set.

normalized:

$$Sim_{col}(\mathcal{M}_q,\ \mathcal{M}_n) = \frac{1}{\sum_1^l w_{l_i}} \sum_{i=1}^{l} w_{l_i} Sim_{col}^{i}(\mathcal{M}_q,\ \mathcal{M}_n). \tag{9}$$

5.4. Fuzzy aggregation

The previous sections introduced two different similarity measures, one extracted from shape information only and the other concerning spatial color distribution over shape. Although the color descriptor reflects also shape information and it is rather effective on objects with similar color distributions, the shape only descriptor is clearly more suitable for comparing regions with similar shape but completely different color structure. Therefore it is very important to have a strategy for synergically combining the two similarity scores. This section proposes a combination strategy based on fuzzy logic.

Let us recall that, given a 3D query model $\mathcal{M}_q$ and a set of candidate models $\mathcal{C} = \{\mathcal{M}_n; n = 1, 2, .., N\}$, the target of the 3D model retrieval system is the identification of the subset of the "relevant" models $\mathcal{C}_R(\mathcal{M}_q) \subset \mathcal{C}$, i.e., of the 3D models belonging to the same class of $\mathcal{M}_q$. To this purpose it is possible to build two fuzzy sets for each query model $\mathcal{M}_q$, one for shape and one for color similarity, denoted as $\mathcal{F}_s(\mathcal{M}_q)$ and $\mathcal{F}_c(\mathcal{M}_q)$ respectively. Each of the two fuzzy sets contains the candidate models $\mathcal{M}_n$ that are supposed to be in the same class of $\mathcal{M}_q$ according to the corresponding similarity measure.

Each element $X_j \in X$ of a fuzzy set $\mathcal{F}$ belongs to $\mathcal{F}$ according to a "degree of membership" $\mu(X_j)$ defined by the so called "membership function" $\mu : X \rightarrow [0, 1]$. In our retrieval scheme the "degree of membership" of a candidate model $\mathcal{M}_n$ to $\mathcal{F}_s(\mathcal{M}_q)$ is defined by membership function $\mu_{s,q}(\mathcal{M}_n)$ that depends on the shape similarity Sim_{col} between the query model $\mathcal{M}_q$ and the candidate model $\mathcal{M}_n$. In a similar way the "degree of membership" of $\mathcal{M}_n$ to $\mathcal{F}_c(\mathcal{M}_q)$ is defined by the membership function $\mu_{c,q}(\mathcal{M}_n)$ which depends on the Sim_{col} measure between the query and candidate object. Formally, fuzzy sets $\mathcal{F}_s(\mathcal{M}_q)$ and $\mathcal{F}_c(\mathcal{M}_q)$ can be denoted as:

$$\mathcal{F}_s(\mathcal{M}_q) = \{\mathcal{C}, \mu_{s,q}(\mathcal{M}_n)\} = \{\mathcal{C}, \mu_s(Sim_{sh}(\mathcal{M}_q,\mathcal{M}_n)\} \tag{10}$$

$$\mathcal{F}_c(\mathcal{M}_q) = \{\mathcal{C}, \mu_{c,q}(\mathcal{M}_n)\} = \{\mathcal{C}, \mu_c(Sim_{col}(\mathcal{M}_q,\mathcal{M}_n)\} \tag{11}$$

Note how the "degree of membership" depends only on the computed similarity measure between the query and candidate model, i.e., there are two membership functions for each query model (one for color and one for shape) but the two functions that map the similarity values into the "degree of membership" are the same for the whole set of candidate models. The dependency of the membership functions from $\mathcal{M}_q$ is only through the similarity values.

The performance of a fuzzy system crucially depends on the definition of the membership functions that map the shape and color similarities into the "degree of membership" to the fuzzy sets. How to compute the two membership functions $\mu_{s,q}(\mathcal{M}_n) = \mu_s(Sim_{sh})$ and $\mu_{c,q}(\mathcal{M}_n) = \mu_c(Sim_{col})$ exploiting a learning stage will be the subject of the next subsection. Assuming that $\mu_{s,q}(\mathcal{M}_n)$ and $\mu_{c,q}(\mathcal{M}_n)$ are known, the proposed joint similarity index between a query model $\mathcal{M}_q$ and a candidate model $\mathcal{M}_n$ exploiting both shape and color information is defined as

$$Sim_{com}(\mathcal{M}_q, \mathcal{M}_n) = \mu_{s,q}(\mathcal{M}_n) \odot \mu_{c,q}(\mathcal{M}_n) \tag{12}$$

The "$\odot$" operator between two fuzzy memberships μ_1 and μ_2 is defined in the following way:

$$\mu_1 \odot \mu_2 = \max\{\mu_1,\ \mu_2\}^{\gamma} \min\{\mu_1,\ \mu_2\}^{1-\gamma} \tag{13}$$

Basically the proposed operator represents a trade-off between the AND and the OR operators and has been inspired by the work in [34]. The rationale behind (12) is that combining μ_s and μ_c by the AND operator (similar objects must have a high score both for color and shape similarity) can be too restrictive but at the same time, by using the OR operator (just one kind of similarity is required) leads to poor precision performance. The parameter $\gamma \in [0, 1]$ moves the "$\odot$" operator towards an AND operator ($\gamma = 1$) or towards an OR operator ($\gamma = 0$). It does not change the relative weight of shape and color. For the experimental results in this paper we set a value of $\gamma = 0.3$ that corresponds to slightly favour the OR operator since it provided the best performances. However the system is not too sensitive to the value of γ. Constant γ is the only parameter of the aggregation procedure since the membership functions are computed once forever in the training phase. A very important feature of the proposed scheme is that there are no parameters controlling the relevance of the color with respect to the shape. This fundamental issue is instead solved through the learning of the membership functions on the training set as explained in the following subsection.

5.5. Learning of the fuzzy membership functions

As already said, the definition of the membership function is a fundamental issue for a fuzzy system. In the proposed retrieval procedure the membership functions μ_s and μ_c are built from a learning stage performed on a training set of 3D models $\mathcal{T} = \{\mathcal{M}^t_k, k = 1, 2, ..., K\}$. The models in the training set are purposely denoted as $\mathcal{M}^t_k$ in order to stress also notationally that they are not the models $\mathcal{M}_n$ of the candidate set and they should not be confused with them. However, as typical of learning procedures, the general properties of the models of the training set should be quite representative of the ones of the models in the candidate set $\mathcal{C}$.

In the proposed procedure, the definition of μ_s and μ_c is inspired by the concept of *precision* in information retrieval, which is the ratio between the relevant documents found with respect to the total number of retrieved documents. In other words, given a fixed number of retrieved documents, the precision is the probability that one of the retrieved document is really a member of the class of the relevant objects: this corresponds to the concept of membership function for a fuzzy set. The actual computation of the membership functions μ_s and μ_c can be made by the procedure of Algorithm 1, which is explained next.

Let us consider first the computation of the shape membership function μ_s. Each of the K models of the training set $\mathcal{T}$ is compared with all the other models of $\mathcal{T}$, i.e. after selecting a query model $\mathcal{M}^t_q, q = 1, .., K$ from the training set we consider as candidate set $\mathcal{C}^t = (\mathcal{T} - \mathcal{M}^t_q)$ and the proposed shape similarity index between $\mathcal{M}^t_q$ and all the models in $\mathcal{C}^t$ is computed according to (4). Let us denote with $s_{qn}, n = 1, .., K$ the shape similarity indexes computed in this way and sorted on the basis of the similarity values, i.e. :

$$s_{qn} = Sim_{sh}(\mathcal{M}^t_q,\ \mathcal{M}^t_n),\ n = 1, .., K \tag{14}$$

where the models have been sorted in order to have $s_{qi} \geq s_{qj}$ if $i < j$. Let us define set

$$S_{s_{qn}} = \{\mathcal{M}^t_j | s_{qj} \geq s_{qn}\} = \{\mathcal{M}^t_j | Sim_{sh}(\mathcal{M}^t_q,\ \mathcal{M}^t_j) \geq s_{qn}\} \tag{15}$$

i.e. $S_{s_{qn}}$ is the set of all the models $\mathcal{M}^t_j$ with shape similarity index greater or equal to s_{qn} with respect to the query model $\mathcal{M}^t_q$ and let $R_{s_{qc}}$ denote the set of the “relevant” objects in $S_{s_{qn}}$ (i.e. the 3D models of $S_{s_{qn}}$ that belong to the same class of $\mathcal{M}^t_q$).

The above procedure requires to use every model in the training set $\mathcal{M}^t_q, q = 1, 2, ..., K$ as query model and to compute the corresponding $S_{qn}, n =$

$1,..,K$ sets. Such sets are used to compute the precision[2] $P(s_{qn}) = |R_{s_{qn}}|/|S_{s_{qn}}|$ as a function of the Sim_{sh} index. Let us stress that this computation is done for each query model $\mathcal{M}_q^t, q = 1,..,K$, hence each model in the training set is used once as a query model and $(K-1)$ times as a candidate model. As explained above the meaning of $P(s_{qn})$ is very close to that of a shape membership function with respect to the specific query model $\mathcal{M}_q^t$. In this way, for each query and candidate couple of models (M_q^t, M_n^t) in the training set, a point $(s_{qn}, P(s_{qn}))$ is obtained on the precision vs. similarity curve corresponding to the selection of all the retrieved objects with similarity equal or greater than s_{qn} with respect to the query model $\mathcal{M}_q^t$.

Points $(s_{qn}, P(s_{qn}))$ cannot be used directly to get the shape membership function since they are sparsely sampled with respect to the abscissa values and because each point $(s_{qn}, P(s_{qn}))$ is intrinsically related to the similarity of every 3D model in the training set with respect to a single query model $\mathcal{M}_q^t$. The sparsity issue can be overcome as follows: for each model $\mathcal{M}_q$ the corresponding set of samples $(s_{qn}, P(s_{qn}))$ is interpolated and resampled over a set of dense and uniform abscissa values $s_{qp} = s_p, p = 1,...,N_p$ identical for every q, obtaining dense curves $(s_p, P(s_{qp}))$ (with $N_p >> K$). Let us observe that notation $s_p = s_{qp}$ is perfectly valid since values s_{qp} are the same for every q.

The other issue, i.e. the previously noted dependence from $\mathcal{M}_q^t$, can be overcome by averaging curves $(s_p, P(s_{qp}))$ over the training set. More precisely the columns of the $K \times N_p$ matrix $P(S_{qp}), q = 1,...,K; p = 1,...,N_p$ are averaged in order to obtain the shape membership function μ_s:

$$\mu_s(s_p) = \frac{1}{K}\sum_{q=1}^{K} P(s_{qp}) \qquad p = 1,..,N_p \tag{16}$$

i.e., $\mu_s(s_p)$ is the average of all the precision vs. similarity curves corresponding to the different models in the training set. As already underlined in Section 5.4 μ_s is the function that maps the shape similarity value into the degree of membership to the fuzzy set. The proposed procedure uses training set $\mathcal{T}$ in order to learn a single membership function to be subsequently used with all the possible query models $\mathcal{M}_q$. Note that the dependency of the degree of membership to the query model is only through the similarity value,

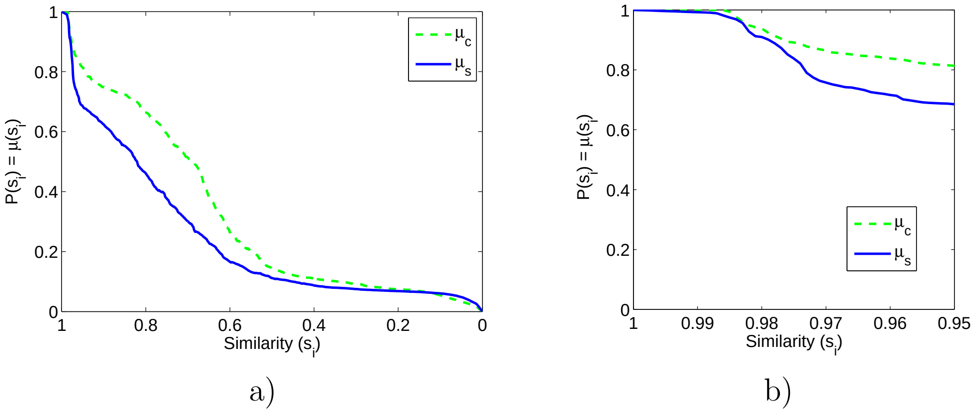
[2] Operator $|A|$ indicates the cardinality of set A.

i.e., $\mu_{s,q}(\mathcal{M}_n) = \mu_s(s_{qn}) = \mu_s(Sim_{sh}(\mathcal{M}_q, \mathcal{M}_n))$. The solid line in Fig. 4 represents the μ_s function used for all the experimental results of this paper. The computed function basically maps the computed shape similarity values to the probability that the model is really relevant for the query. In this way it allows to combine together different similarity measures even if they are based on completely different clues and have a different measurement scale. Note how the approach is here used to combine shape and color but can in principle be used to combine different similarity measures, referring to any characteristic.

As indicated in Algorithm 1 the color membership function μ_c is obtained by the same operations required to compute μ_s but using the color similarity measure Sim_{col} index instead of the shape one. Fig. 4 refers to the the training set used for the experimental results of this paper and shows the resulting membership functions μ_s and μ_c represented by the solid and dotted line respectively.

Algorithm 1: Learning of μ_s and μ_c from the training set.

```
Assume a training set made by $K$ 3D models.
for $q = 1$ to K do
    for $n = 1$ to K do
        Compute $s_{qn} = Sim_{sh}(\mathcal{M}^t_q,\ \mathcal{M}^t_n)$
    end for
    Sort the $s_{qn}$ values from the most similar to the least similar
    for $n = 1$ to K do
        Compute sets $S_{s_{qn}}$ and $R_{s_{qn}}$ and evaluate precision
        $P(s_{qn}) = |R_{s_{qn}}|/|S_{s_{qn}}|$
    end for
    Interpolate points $(s_{qn}, P(s_{qn})), n = 1, .., K$ to
    $(s_p, P(s_{qp})), p = 1, .., N_p$
end for
Compute $\mu_s(s_p) = \frac{1}{K}\sum_{q=1}^{K} P(s_{qp}) \qquad p = 1, .., N_p$
$\mu_c$ can be computed with the same procedure but with
    $s_{qn} = Sim_{col}((\mathcal{M}_q,\ \mathcal{M}_n), q = 1, .., K, n = 1, ..., K$
```



a) b)

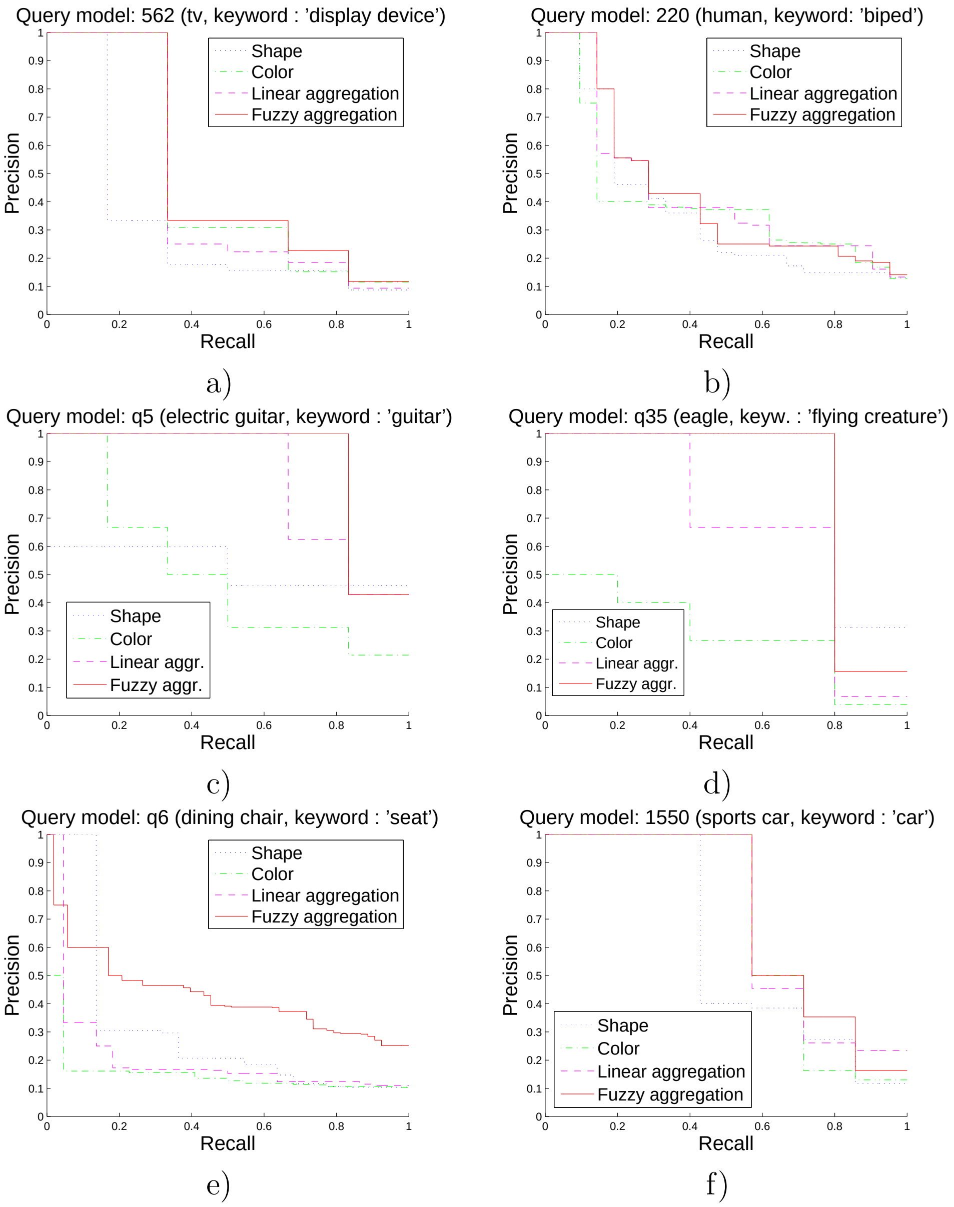
Figure 4: a) Membership functions μ_s and μ_c obtained from the training set and used for the experimental results; b) enlarged detail of the region corresponding to similarity values between 0.95 and 1 region.

6. Experimental results

In order to evaluate the performance of the proposed 3D model retrieval method with a standard dataset we used a modified subset of the *Princeton Shape Benchmark* (PSB) 3D model database [33]. The dataset $\mathcal{C}$ of the candidate models used in the experiments is made by 250 models from the PSB randomly selected, equally subdivided among the classes of the original database and suitably texturized. The query models $\mathcal{M}_q$ were selected both from the PSB dataset and from the SHREC 2008 queryset [35]. Color information was also added to the query models. A different set of models also randomly taken from the PSB and texturized was used as training set $\mathcal{T}$ to build the μ_s and μ_c functions as described in Subsection 5.5. The tests were made by comparing each query model $\mathcal{M}_q$ with all the other models in the candidate models database $\mathcal{C}$, and by ordering the results on the basis of the similarity values. The resulting classifications were then evaluated using the hierarchical subdivision originally provided by the PSB, i.e., by considering as "ground truth" the second and third hierarchical levels of the query models (first and second if the third was not defined). In this manner the candidate models were divided as *marginally relevant* (when the second hierarchical level agrees with the query one) or *highly relevant* (when also the third hierarchical level agrees with the query one). All the 3D models either highly or marginally relevant are then considered as relevant. Several performance measures such as *precision vs. recall* trend, *average precision*, *average dynamic recall* [36], *Mean first and second tier*, *cumulated gain* [37] and *E-measure* were considered as evaluation metrics.

Fig. 5 shows some sample results exemplifying the limits of the usage of shape information only or color information only and the benefits of combining both color and shape information for 3D model retrieval. Namely, Fig. 5 reports the performance of the proposed method on some 3D models both from the PSB dataset and from the queryset of the SHREC 2008 contest[3] [35]. Fig. 6 shows the thumbnails of the models returned by the query in the human 3D model case. The blue dotted line in Fig. 5 shows the precision-recall figures based on shape information only according to the Sim_{sh} measure of Equation (4) while the green dot dashed line refers to the retrieval based on color information alone according to the Sim_{col} measure of Equation (9). Both methods used by themselves are not completely satisfactory. 3D retrieval based on shape information only, as expected, ignores all the additional clues given by color information. On the converse color spin-images, while capable of matching the different regions of the objects from their spatial color distribution, may fail in the case of heavily texturized objects with many irregularly distributed colors or in case of objects with similar shape but completely different color distribution. The magenta dashed line instead refers to the results of the weighted linear combination of the two measures (the weights have been optimized from a training phase performed on the same set $\mathcal{T}$ used for the training of the fuzzy recombination scheme). It is possible to note that the results are better than the ones obtained with color or shape information alone but still not completely satisfactory, e.g., in Fig. 5a and Fig. 5e the poor results of shape and color retrieval respectively affect also the results of the linear combination scheme. Finally the red line refers to the fuzzy recombination of color and shape information according to the method of Section 5.4. The figure shows how the proposed fuzzy scheme returns substantially better results than exploiting only color or only shape information and also outperforms the simple linear combination scheme. In particular it can be observed from Fig. 5c and Fig. 5e that the results are much better than the ones of each descriptor type alone and that the poor performance of the color descriptor on the *eagle* model in

[3] We used the 2008 edition of the SHREC contest and not the most recent edition [8] since it was the last one based on the PSB dataset for generic models. This fact allows a direct comparison of the results of other methods with ours.



Query model: 562 (tv, keyword : 'display device')

Query model: 220 (human, keyword: 'biped')

Query model: q5 (electric guitar, keyword : 'guitar')

Query model: q35 (eagle, keyw. : 'flying creature')

Query model: q6 (dining chair, keyword : 'seat')

Query model: 1550 (sports car, keyword : 'car')

a)

b)

c)

d)

e)

f)

Figure 5: Precision vs. recall trends for some sample models: a) *television* model; b) *human* model; c) *electric guitar* model; d) *eagle* model; e) *chair* model; f) *car* model. The plots compares the precision vs. recall figures obtained with shape descriptors only (blue dotted line), color descriptor only (green dashdot line), linear combination of the two measures (magenta dashed line) and finally the aggregation of the two measures by the proposed fuzzy scheme (red solid line).

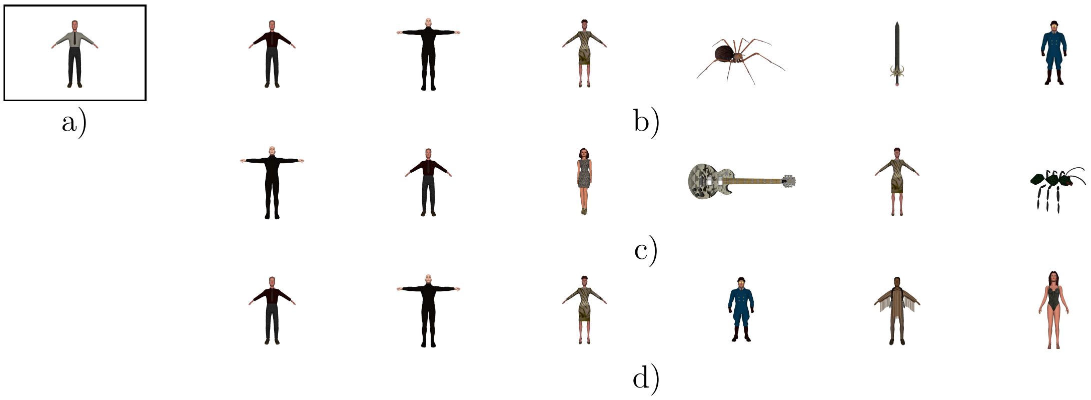
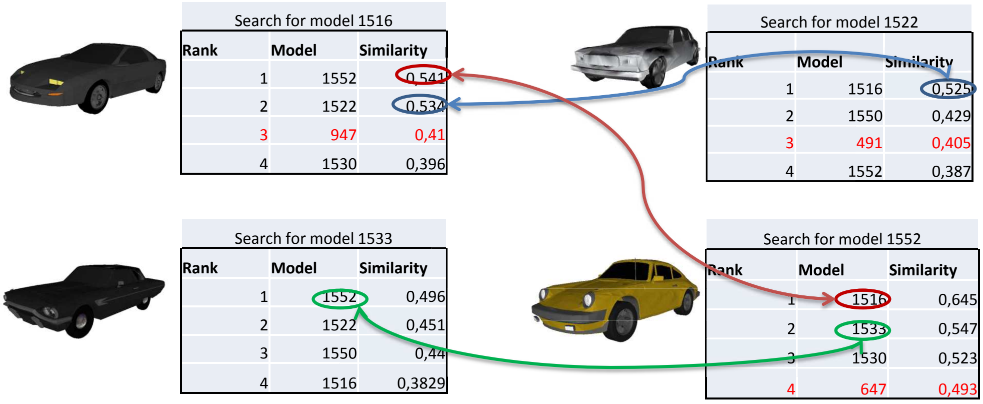


Figure 6: Retrieval results for a human model model (texturized version of model 222 of the PSB). a) The query model; b) Results using shape information alone; c) Results using color information alone; d) Combined use of color and shape information.

Fig. 5d does not impact the fuzzy aggregation results. In general the fuzzy aggregation scheme is quite robust also in situations where one of the two clues does not work as well as the other, while the simple linear combination scheme is less robust in these situations.

Fig. 7 shows another set of results for some sample queries in the *car* category. Besides showing that our approach is quite effective also on this category, the purpose of this figure is to show how the results of the queries performed on similar objects are consistent (i.e., if model $\mathcal{M}_1$ is similar to $\mathcal{M}_2$ also the opposite is true). As shown by the data of Fig. 7, the fact that the proposed similarity measure is not symmetric has a minor impact on the results as discussed in Subsection 5.1.

Fig. 8 shows the performances on some query models taken from the SHREC 2008 contest [35]. The plots in Fig. 8 compare the precision vs. recall figures of our method (denoted as "ColorSI" in the plots) against the light field algorithm [22], and some methods from the SHREC2008 contest. Note how all the competing approaches exploit only shape data while the proposed can take advantage of also color information, however the scope of this paper is exactly to show that color information can be effectively used to improve retrieval performances. The light field results have been obtained with the implementation available on the web and the ones of the SHREC



Search for model 1516

Rank	Model	Similarity
1	1552	0,541
2	1522	0,534
3	947	0,41
4	1530	0,396

Search for model 1522

Rank	Model	Similarity
1	1516	0,525
2	1550	0,429
3	491	0,405
4	1552	0,387

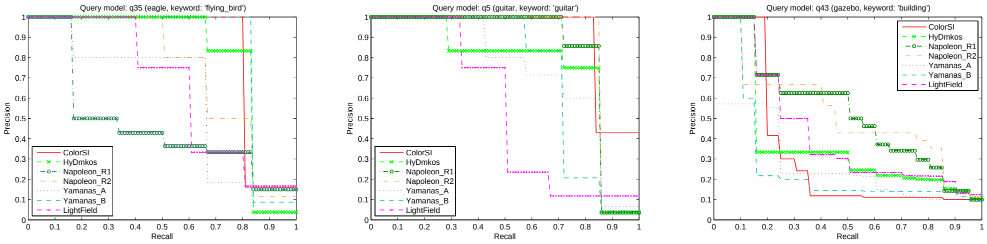
Search for model 1533

Rank	Model	Similarity
1	1552	0,496
2	1522	0,451
3	1550	0,44
4	1516	0,3829

Search for model 1552

Rank	Model	Similarity
1	1516	0,645
2	1533	0,547
3	1530	0,523
4	647	0,493

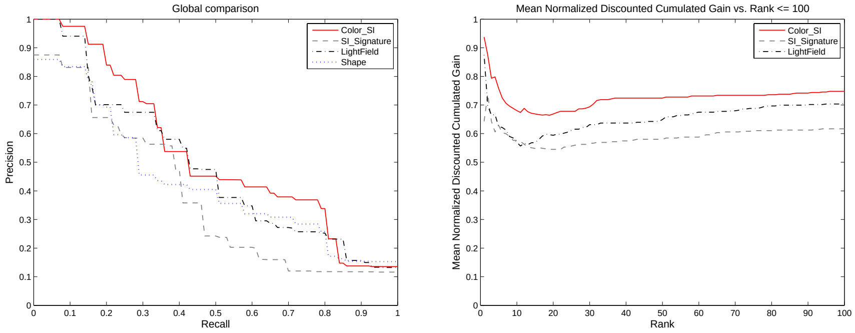
Figure 7: Results for some sample queries on similar objects in the car category. The numbers in black correspond to relevant objects belonging to the car category, while the red ones belong to other categories.



a) b) c)

Figure 8: Precision vs. recall trends for some example queries from the SHREC 2008 query set: a) *eagle* model (query 35); b) *guitar* model (query 5); c) *gazebo* model (query 43). The compared methods are: our method (ColorSI), [38] (HyDmkos), [39] (Napoleon_R1 and Napoleon_R2), [40] (Yamanas_A), [41] (Yamanas_B) and the light field based method of [22] (LightField).

methods come from the results file on the SHREC website[4]. Fig. 8 shows that the proposed method typically performs better than most of the other content-based retrieval methods (in fact it is outperformed on the *eagle* model only by [41] and on the *guitar* model only by [39]). The limits of the proposed method are well exemplified by its performance on the *gazebo* model reported in Fig. 8c. In this case the dataset contains 3 gazebos (*highly relevant*) and 20 buildings (*relevant*). The proposed scheme outperforms the other methods in retrieving all the other gazebos, that indeed have a similar color structure, while its performance is not particularly impressive in retrieving the other buildings, which have a rather different color structure than those of the gazebos.



a) b)

Figure 9: Comparison of our approach, [22] and [17] over the query models set. a) Mean interpolated-precision vs. recall trends. b) Mean normalized discounted cumulated gain trends (the trends refer to the first 100 elements of the ranked lists).

Fig. 9a shows the precision vs. recall average on a set of queries on the PSB. The plot compares the proposed method against those for which we had an available implementation, i.e., the light field descriptor based method of [22] and the spin image retrieval scheme of [17] (the results of this method have been kindly provided by the authors). The normalized discounted cumulated gain trends for the same set of queries are shown in Fig. 9b. On the average our method is slightly better than the light field descriptor. From Fig. 9a it is also worth noting that the performance of the proposed approach when using shape information only is rather similar to

[4] To compare our results with the methods of the contest, the result files of the available methods were filtered in order to consider only the models included in our subset.

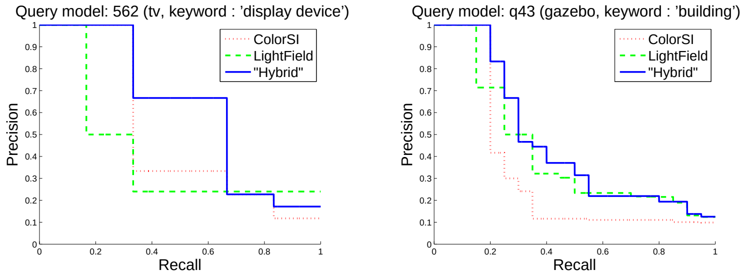
that of [17], consistently with the fact that both methods are based on the same type of descriptor. Instead the performance of the proposed method is clearly better than that of [17] when both shape and color information are combined together. The performance difference between the two methods is a good indication of the improvement color information may bring to 3D retrieval (the method of [17] exploits shape data only as most methods in the literature).

Table 2 reports some different performance measures for the three approaches, namely mean average precision, first and second tier values, mean dynamic average recall and E-measure. Note that some of the measures have been computed both by considering relevant all the models that are highly or only marginally relevant (the ones denoted with *relevant*) as in the rest of the result section and by considering just the highly relevant ones (*highly rel.*). The results of Table 2 confirm the advantages of jointly using shape and color information for 3D model retrieval, indeed the proposed approach is able to outperform both the compared solutions in most of the performance measures.

Performance measure	LF [22]	SI [17]	ColorSI (our method)
Mean Average Precision (highly relevant)	0.607	0.614	**0.649**
Mean Average Precision (relevant)	0.477	0.367	**0.525**
Mean First Tier (highly relevant)	54.16	**60.83**	52.31
Mean First Tier (relevant)	43.72	35.37	**48.87**
Mean Second Tier (highly relevant)	34.72	33.75	**36.34**
Mean Second Tier (relevant)	26.37	22.42	**29.06**
Mean Dynamic Average Recall	0.534	0.465	**0.627**
E-Measure (highly relevant)	0.178	0.111	**0.190**
E-Measure (relevant)	**0.322**	0.253	0.319

Table 2: Performance comparison between the approach of [22] (denoted with LF), the one of [17] (denoted with SI) and proposed approach (ColorSI)

The proposed fuzzy aggregation scheme can in principle combine results from any similarity measure. An interesting question that arise at this point is which results it is possible to obtain if the proposed color descriptor is combined with state-of-the-art shape descriptors from the literature. This aspect was tested on an “hybrid” approach where the shape similarity measure is computed with the Light Field method of [22] and the color measure is based on the color descriptor introduced in this paper. In this way by using two completely different descriptors it is possible to obtain good results even on objects that are particularly difficult to represent with spin-images. Fig. 10 shows the results on a couple of sample models obtained exploiting this approach. Fig. 10a refers to a *television* model and shows how the proposed fuzzy combination (denoted with “hybrid” in the plots) performs better than each of the two approaches alone. Fig. 10b shows the result on the *gazebo* model. Lets recall from Fig. 8c that the *gazebo* is a quite critical model for our approach, which is able to retrieve the other gazebos correctly ranking in the first positions but fails to retrieve most of the other buildings. In spite the performance of our approach (ColorSI in the plots) on this object is inferior with respect to that of [22], it is worth noting that the “hybrid” method obtained by the fuzzy recombination of the light field shape descriptor and the color descriptor introduced in this paper outperforms 3D retrieval based on the shape information only by way of [22]. Furthermore the proposed “hybrid” method inherits from our color descriptor the capability of ranking the gazebos in the first positions and from [22] the ability of giving good results within all the *building* class. This test demonstrates the general effectiveness of the usage of color information and of the proposed fuzzy recombination scheme for textured 3D model retrieval. It shows also that combining with the proposed color descriptor shape descriptors based on concepts different than spin-images gives good results on objects not well suited to be represented by spin-images.



a) b)

Figure 10: Precision vs. recall trends for a) a *television* model; b) a *gazebo* model. The plots compares the results of the proposed method that exploits spin-images for shape and color descriptors (red dotted line), of the light field based method of [22] (green dashed line) and of the aggregation of the shape similarity measure of [22] together with the proposed color descriptor based on spin-images (blue solid line).

As a final remark, it is worth pointing out is that the proposed approach has also the advantage of being rather fast: the comparison between the descriptors associated to two different models on the average takes only $0.01s$ on a standard desktop PC[5] with the current implementation that has not been optimized and does not exploit GPU or parallel architectures. In spite of fast execution time, searching in large repositories requires a time proportional to the number of models which may become too large in some applications. Besides simply optimizing the code a possible solution for this issue is a two steps procedure motivated by the fact that in most retrieval applications we are interested in finding the most similar objects and the eigenvalues usually decay quite fast. In the first step only the first eigenvectors are compared since they correspond to a large percentage of the similarity score. If the similarity score is above a threshold the procedure continues with the second step where all the other eigenvectors are compared, otherwise the comparison will stop since the similarity score will never reach a high value. This corresponds to a first raw analysis of the global shape followed by the analysis of smaller structures only if the two general structures pass a first similarity check. Different versions of this *hierarchical* comparison approach (e.g. performing this check multiple times) can be adopted in order to build an efficient search engine based on the proposed method. Other possible approaches for efficient searching in spin-image descriptors that avoid the comparison of the query object with all the candidate ones have been presented in [30].

The descriptor computation is more computationally demanding and it takes from $5s$ to $45s$ depending on the 3D model to be analyzed[6]. However this task is usually done just once for each model and then the descriptor can be stored. Furthermore there exist efficient solutions for the computation of spin-image descriptors (that is the most time-demanding step) which have not been exploited in this work. With the parameters used for the experimental results the size of each descriptor can vary from $10kB$ to $100kB$ with an average size of about $30kB$. If smaller descriptors are needed, e.g., for searching in large databases, the threshold e_{ms} can be increased in order to to reduce the cutoff values, thus reducing descriptor size at the price of a

[5] The tests has been done on a DELL XPS420 PC with an Intel Core2 Q6600 processor and 4 Gb of RAM. Note how even if this computer has a quad-core CPU the current implementation is not able to exploit multi-core architectures.

[6] These times refer to the PSB models on the same PC used for the descriptor comparison.

small decrease in the retrieval accuracy. Fig. 11 shows an example of the impact of increasing the value of e_{ms}. It shows the retrieval results referring to the same *car* objects of Fig. 7 for 4 different values of e_{ms}, i.e. $e_{ms} = 0,05$ as used in the rest of the paper, $e_{ms} = 0,075$, $e_{ms} = 0,1$ and $e_{ms} = 0,2$, respectively corresponding to average descriptor sizes of $30kB$, $25kB$, $20kB$ and $12kB$. For example by doubling e_{ms} from $0,05$ to $0,1$ it is possible to reduce the descriptor size and comparison time of about 33% with a negligible impact on the results. A further increase of e_{ms} to $0,2$ allows a large further descriptor size reduction but at some accuracy loss.

	Rank	e_{ms}=0,05	e_{ms}=0,075	e_{ms}=0,1	e_{ms}=0,2
 Model #: 1552	1	1516	1516	1516	1516
	2	1533	1533	1533	1533
	3	1530	1530	1530	647
	4	647	647	647	1522
	5	1522	1522	1522	1528

	Rank	e_{ms}=0,05	e_{ms}=0,075	e_{ms}=0,1	e_{ms}=0,2
 Model #: 1516	1	1552	1522	1522	1522
	2	1522	1552	1552	1552
	3	947	947	947	946
	4	1530	835	835	835
	5	469	1530	1530	947

	Rank	e_{ms}=0,05	e_{ms}=0,075	e_{ms}=0,1	e_{ms}=0,2
 Model #: 1533	1	1552	1552	1552	1552
	2	1522	1522	1522	1522
	3	1550	1550	1550	1550
	4	1516	1516	1516	1516
	5	1528	1528	1528	1528

	Rank	e_{ms} = 0,05	e_{ms} = 0,075	e_{ms} = 0,1	e_{ms} = 0,2
 Model #: 1522	1	1516	1516	1516	946
	2	1550	1550	1550	1516
	3	491	491	946	1550
	4	1552	946	491	491
	5	107	107	107	107

Figure 11: Results for some sample queries on similar objects in the car category with descriptors built with different values of e_{ms}. The numbers in black correspond to relevant objects belonging to the car category, while the red ones belong to other categories.

7. Conclusions and future work

This paper proposes a novel content-based retrieval scheme for colored 3D models with a number of specific points of interest. First of all, the idea that color information, when properly processed, may improve content-based retrieval of 3D models, is a very innovative and intriguing topic. A second contribution of the paper is its original color descriptor, which captures color information in a structure inspired by the spin-images 3D shape descriptor and which proves able to capture the shape of the color regions over the 3D model instead of just the color differences between different 3D models. This non trivial feature is the key for recognizing as similar also models with different colors but with the same distribution of the color over the shape. A further relevant contribution of the paper is the proposed fuzzy scheme for recombining shape and color similarity into a single measure. The crucial issue for any fuzzy scheme of defining the membership functions is solved by a learning procedure on a training set. Such a procedure is motivated by a reinterpretation of the meaning of the precision vs. recall figures in terms of fuzzy logic.

The produced experimental evidence confirms that the combined use of color and shape information allows to obtain better performance than each information type alone. More precisely the proposed method is able to outperform state-of-the-art 3D retrieval solutions from the literature and the SHREC content retrieval contest, with particularly satisfactory performances on categories where the color structure presents distinctive patterns. The proposed approach has also the ability of ranking highly relevant models in the first positions of the retrieval list, a rather useful feature in many applications, such as search engines. In summary the contribution of this work can be considered twofold: the idea that color information, properly processed, may improve content-based retrieval of 3D models and the experimental confirmation of such a possibility within a concrete method.

The research direction indicated by this work can be improved in many ways. First of all various aspects of the proposed method are worth further exploration. For instance, spin-images are particularly sensible to mesh sampling and *ad-hoc* remeshing solutions targeted to these descriptors would be very useful. Concerning color information there are two main aspects worth further investigation. The first is the study of quantization schemes with a better trade-off between descriptor accuracy and number of quantization bins than the proposed quite coarse quantization scheme. The second is the exploration of color descriptors based not only on the vertex colors but also on some statistical properties in the neighbourhood of each vertex in order to improve retrieval performance on highly textured objects with complex color patterns.

We also plan to extend the specific approach of this work based on spin-images to other existing shape descriptors and to develop new ones specifically conceived for embedding shape and color information. Indeed any improvement of the shape retrieval method adopted within the shape and color 3D retrieval system will allow to improve the overall performance. Another intriguing possibility, open by the generality of the proposed fuzzy recombination scheme, is the joint usage of different shape descriptors in order to strengthen shape retrieval performance.

An interesting characteristic of the proposed fuzzy recombination scheme is its generality, indeed it could be used not only to combine the proposed color descriptor with other shape retrieval schemes but also to combine together several shape and color descriptors in order to improve retrieval performances. The exploration of different solutions in this direction opens interesting possibilities for the construction of more effective and accurate 3D model retrieval systems.

Finally in order to allow the construction of practical applications based on the proposed approach we are planning to optimize the current implementation including in it efficient algorithms for the descriptor computation and smarter searching techniques. In particular we will focus on searching strategies that will allow to avoid the comparison of the query model with all the candidate models in order to better scale to large databases.

Acknowledgment

The authors would like to thank Alessio Astolfi and Ivano Calabrese for their early contribution to the development of the retrieval software. We are also very grateful to Prof. Pietro Pala and Prof. Alberto Del Bimbo for providing the performance measures of their retrieval scheme used for the comparisons in the experimental results.

References

[1] Google, Google 3d warehouse, 2012. Accessed 30 July 2012.

[2] G. Cortelazzo, N. Orio, Retrieval of colored 3d models, in: Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission, pp. 986 –993.

[3] A. E. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3d scenes, IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (1999) 433–449.

[4] J. W. Tangelder, R. C. Veltkamp, A survey of content based 3d shape retrieval methods, Multimedia Tools Appl. 39 (2008) 441–471.

[5] Z. Qin, J. Jia, J. Qin, Content based 3d model retrieval: A survey, in: Proceedings of the International Workshop on Content-Based Multimedia Indexing (CBMI 2008), pp. 249 –256.

[6] A. Del Bimbo, P. Pala, Content-based retrieval of 3d models, ACM Transactions on Multimedia Computing Communications and Applications 2 (2006) 20–43.

[7] O. Van Kaick, H. Zhang, G. Hamarneh, D. Cohen-Or, A survey on shape correspondence, in: Proc. of Eurographics State-of-the-art Report, pp. 1–22.

[8] B. Li, A. Godil, M. Aono, X. Bai, T. Furuya, L. Li, R. Lpez-Sastre, H. Johan, R. Ohbuchi, C. Redondo-Cabrera, A. Tatsuma, T. Yanagimachi, S. Zhang, Shrec'12 track: Generic 3d shape retrieval, in: Proceedings of Eurographics Workshop on 3D Object Retrieval 2012 (3DOR 2012), Eurographics Association, 2012, pp. 119–126.

[9] E. Paquet, M. Rioux, A. Murching, T. Naveen, A. Tabatabai, Description of shape information for 2-d and 3-d objects, Signal Processing: Image Communication 16 (2000) 103 – 122.

[10] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, D.and Jacobs, A search engine for 3d models, ACM Transactions on Graphics (2003) 83–105.

[11] H. Laga, M. Nakajima, K. Chihara, Discriminative spherical wavelet features for content-based 3d model retrieval, International Journal of Shape Modeling (IJSM) 13 (2007) 51–72.

[12] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, Shape distributions, ACM Transactions on Graphics 21 (2002) 807–832.

[13] J.-L. Shih, C.-H. Lee, J. Wang, 3d object retrieval system based on grid d2, Electronics Letters 41 (2005) 179 – 181.

[14] Y. Gao, Q. Dai, N.-Y. Zhang, 3d model comparison using spatial structure circular descriptor, Pattern Recognition 43 (2010) 1142 – 1151.

[15] A. Mademlis, P. Daras, D. Tzovaras, M. G. Strintzis, 3d object retrieval using the 3d shape impact descriptor, Pattern Recognition 42 (2009) 2447 – 2459.

[16] A. M. Bronstein, M. M. Bronstein, L. J. Guibas, M. Ovsjanikov, Shape google: Geometric words and expressions for invariant shape retrieval, ACM Transactions on Graphics 30 (2011) 1:1–1:20.

[17] J. Assfalg, M. Bertini, A. Bimbo, P. Pala, Content-based retrieval of 3-d objects using spin image signatures, IEEE Transactions on Multimedia 9 (2007) 589 –599.

[18] P. A. De Alarcón, A. D. Pascual-Montano, J. M. Carazo, Spin images and neural networks for efficient content-based retrieval in 3d object databases, Image and Video Retrieval (2002) 225–234.

[19] Y. Liu, H. Zha, H. Qin, The generalized shape distributions for shape matching and analysis, in: Proceedings of the IEEE International Conference on Shape Modeling and Applications (SMI '06), IEEE Computer Society, Washington, DC, USA, 2006, p. 16.

[20] Y. Gao, J. Tang, H. Li, Q. Dai, N. Zhang, View-based 3d model retrieval with probabilistic graph model, Neurocomputing 73 (2010) 1900 – 1905.

[21] P. Daras, A. Axenopoulos, A compact multi-view descriptor for 3d object retrieval, in: Proceedings of the 7th International Workshop on Content-Based Multimedia Indexing (CBMI '09), pp. 115 –119.

[22] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, M. Ouhyoung, On visual similarity based 3d model retrieval, in: Computer Graphics Forum (EUROGRAPHICS'03), volume 22, pp. 223–232.

[23] A. Makadia, K. Daniilidis, Spherical correlation of visual representations for3dmodelretrieval, International Journal of Computer Vision 89 (2010) 193–210.

[24] M. Chaouch, A. Verroust-Blondet, A new descriptor for 2d depth image indexing and 3d model retrieval, in: Proceedings of IEEE International Conference on Image Processing (ICIP '07), volume 6, pp. VI –373 –VI –376.

[25] P. Daras, A. Axenopoulos, G. Litos, Investigating the effects of multiple factors towards more accurate 3-d object retrieval, IEEE Transactions on Multimedia 14 (2012) 374 –388.

[26] M. Suzuki, A web-based retrieval system for 3d polygonal models, in: Proceedings of Joint 9th IFSA World Congress and 20th NAFIPS International Conference, volume 4, pp. 2271 –2276 vol.4.

[27] C. R. Ruiz, R. Cabredo, L. J. Monteverde, Z. Huang, Combining shape and color for retrieval of 3d models, in: Proc. Fifth Int. Joint Conf. INC, IMS and IDC NCM '09, pp. 1295–1300.

[28] Y.-J. Liu, Y.-F. Zheng, L. Lv, Y.-M. Xuan, X.-L. Fu, 3d model retrieval based on color + geometry signatures, The Visual Computer 28 (2012) 75–86.

[29] N. Brusco, M. Andreetto, A. Giorgi, G. Cortelazzo, 3d registration by textured spin-images, in: Proceedings of 5th International Conference on 3-D Digital Imaging and Modeling (3DIM '05), pp. 262 – 269.

[30] A. E. Johnson, Spin-images: A representation for 3-d surface matching, Ph.D. thesis, Carnegie Mellon University, 1997.

[31] A. E. Johnson, M. Hebert, Control of polygonal mesh resolution for 3-d computer vision, Graphical Models and Image Processing 60 (1998) 261 – 285.

[32] E. Albuz, E. Kocalar, A. Khokhar, Quantized cielab* space and encoded spatial structure for scalable indexing of large color image archives, in: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2000), volume 6, pp. 1995 –1998 vol.4.

[33] P. Shilane, P. Min, M. Kazhdan, T. Funkhouser, The princeton shape benchmark, in: Proceedings of Shape Modeling International, Genova, Italy.

[34] P. Androutsos, A. Kushki, K. Plataniotis, A. Venetsanopoulos, Aggregation of color and shape features for hybrid query generation in content based visual information retrieval, Signal Processing 85 (2005) 385 – 393.

[35] R. C. Veltkamp, F. B. Ter Haar, Shape retrieval contest (shrec) 2008, in: Proceedings of Shape Modeling and Applications, 2008 (SMI 2008), pp. 215–216.

[36] R. Typke, R. Veltkamp, F. Wiering, A measure for evaluating retrieval techniques based on partially ordered ground truth lists, Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '06) 0 (2006) 1793–1796.

[37] K. Järvelin, J. Kekäläinen, Cumulated gain-based evaluation of ir techniques, ACM Transactions on Information Systems 20 (2002) 422–446.

[38] P. Papadakis, l. Pratikakis, S. Perantonis, T. Theoharis, G. Passalis, Shrec08 entry: 2d/3d hybrid, in: Proceedings of Shape Modeling and Applications, 2008 (SMI 2008), pp. 247 – 248.

[39] T. Napoleon, T. Adamek, F. Schmitt, O. N.E., Shrec08 entry: Multiview 3d retrieval using multi- scale contour representation, in: Proceedings of Shape Modeling and Applications, 2008 (SMI 2008).

[40] A. Yamamoto, M. Tezuka, T. Shimizu, R. Ohbuchi, Shrec08 entry: Semi-supervised learning for semantic 3d model retrieval, in: Proceedings of Shape Modeling and Applications, 2008 (SMI 2008), pp. 241 – 243.

[41] K. Osada, T. Furuya, R. Ohbuchi, Shrec08 entry: Local volumetric features for 3d model retrieval, in: Proceedings of Shape Modeling and Applications, 2008 (SMI 2008), pp. 245 – 246.