

Article

## A New Efficient Optimal 2D Views Selection Method Based on Pivot Selection Techniques for 3D Indexing and Retrieval

Hassan Silkan <sup>1,\*</sup> and Youssef Hanyf <sup>2</sup>

<sup>1</sup> Laboratoire d'informatique, Mathématiques et leurs Applications, El Jadida 24000, Morocco

<sup>2</sup> Laboratoire LAMAPI, Faculté des Sciences, Université Chouaib Doukkali, El Jadida 24000, Morocco; E-Mail: [youssef.hanyf@gmail.com](mailto:youssef.hanyf@gmail.com)

\* Author to whom correspondence should be addressed; E-Mail: [silkan\\_h@yahoo.fr](mailto:silkan_h@yahoo.fr); Tel.: +212-67-4984-646.

Academic Editor: Ahmed El Oualkadi

Received: 8 June 2015 / Accepted: 9 October 2015 / Published: 20 October 2015

---

**Abstract:** In this paper, we propose a new method for 2D/3D object indexing and retrieval. The principle consists of an automatic selection of optimal views by using an incremental algorithm based on pivot selection techniques for proximity searching in metric spaces. The selected views are afterward described by four well-established descriptors from the MPEG-7 standard, namely: the color structure descriptor (CSD), the scalable color descriptor (SCD), the edge histogram descriptor (EHD) and the color layout descriptor (CLD). We present our results on two databases: The Amsterdam Library of Images (ALOI-1000), consisting of 72,000 color images of views, and the Columbia Object Image Library (COIL-100), consisting of 7200 color images of views. The results prove the performance of the developed method and its superiority over the  $k$ -means algorithm and the automatic selection of optimal views proposed by Mokhtarian *et al.*

**Keywords:** 2D views selection; pivot-based indexing; 2D/3D indexing methods; 2D color descriptors;  $k$ -means; curvature scale space

---

### 1. Introduction

Improvements in 3D scanner technology and the availability of 3D models distributed over the Internet are both contributing to create large databases of this type of multimedia data. Searching 3D

databases by content has many promising applications in domains, like CAD, medicine, molecular biology and entertainment.

The major challenge for the retrieval system is how to extract proper features to represent 3D models and search similar models using these descriptors. The existing 3D object retrieval methods can be classified into four main categories: histogram based, transform based, graph based and view based. This latter considers the 3D shape as a collection of 2D projections taken either from canonical or non-canonical viewpoints. Each view is then described by 2D descriptors [1], like Fourier descriptors [2], Zernike moments [3] and extreme curvature scale space [4].

While most 3D object representations are complicated and inefficient: conventional multi-view representations are based on a large number of views and cannot be used in many applications, such as retrieval from large databases [5–7]. In this article, we address the problem of the choice of the optimal number of views by applying an efficient method based on pivot selection techniques for proximity searching in metric spaces [8]. Indeed, it is well known that the pivots are a subset of objects in the database that are used to speed up the search. The position of each pivot with respect to each other and to the rest of the objects in the database determines the index capacity for discarding objects from the result. Many works have proposed different ways of estimating the effectiveness of a pivot or set of pivots. Works like [9–11] indicated that good pivots should be far from each other and also far from the rest of the objects of the database. Bustos *et al.* [8] defined a formal criterion for comparing the effectiveness of two sets of pivots of the same size.

From the above works, we can deduce two important remarks: first, the good pivots are far away from the rest of the objects in the metric space, and good pivots are also far away from each other [12]. Second, the authors in [8] suggest an efficiency criterion that compares two sets of pivots and designates the better of the two. To take into account these observations, it will be of great benefit to integrate the pivot-based algorithm in 2D/3D search; for that, we propose a new method for 2D/3D object indexing and retrieval based on the pivot selection techniques for proximity searching in metric spaces.

The rest of the paper is structured as follows: After related work in Section 2, we present in Section 3 a new method to select the optimal number of views of 3D objects. It consists of using the pivot selection techniques for proximity searching in metric spaces. A metric distance used to compare objects is introduced in Section 4. In Section 5, several results are shown. Finally, the conclusion and future investigations are discussed in the last section.

## 2. Related Literature

View-based 3D object recognition methods make use of viewer-centered object representations where the set of possible appearances of a 3D object is stored as a collection of 2D images: the major problem in this case is that there is potentially an infinite number of possible viewpoints that induce an infinite number of object appearances. To cope with the huge number of viewpoints, two approaches were used to sample a viewpoint and to choose the number of characteristic views: methods using a fixed number of views and methods using a dynamic number of views.

### 2.1. Methods with a Fixed Number of Views

These methods select a fixed number of views, independently of the complexity of 3D model to be indexed. Many methods are proposed in the literature. In [13], the light field descriptor (LFD) was introduced by Chen *et al.* In their approach, ten silhouette images are taken from 10 viewing angles distributed evenly on a dodecahedron. To extract the features of the silhouette images, they used Zernike moments and Fourier descriptors. To calculate the dissimilarity, they found the minimal dissimilarity obtained from rotating the viewing spheres of one light field descriptor relative to the other light field descriptor. Experiments indicate that LFD performs better than spherical harmonics descriptors [14].

Funkhouser *et al.* [15] have a view-based 3D shape descriptor in their 3D model search engine to provide a 2D sketch query interface. The descriptor uses 13 views defined by a bounding cube. Viewpoints are taken at the center of three faces, the four top corners and the middle of six edges of that cube. The approach uses a limited number of views, so its retrieval precision is relatively low.

Chaouch and Verroust-Blondet [16] propose a method where a 3D model is projected to the faces of its bounding box, resulting in six depth buffers. Each depth buffer is then decomposed into a set of horizontal and vertical depth lines that are converted to state sequences, which describe the change in depth at neighboring pixels.

In [17], Ramezani and Ebrahimnezhad provide an effective new 3D model retrieval system based on the Poisson equation. The  $k$  optimal 2D views of 3D object are obtained by using the  $k$ -means clustering method.

Presenting a 3D object with a fixed number of 2D views can lead to some major limitations that depend on the 3D shape complexity. The first one, when the 3D model is complex and contains more information, leads to the problem of under views estimation. In the opposite case, if the 3D model does not have a complex structure, it can lead to another problem of over views estimation.

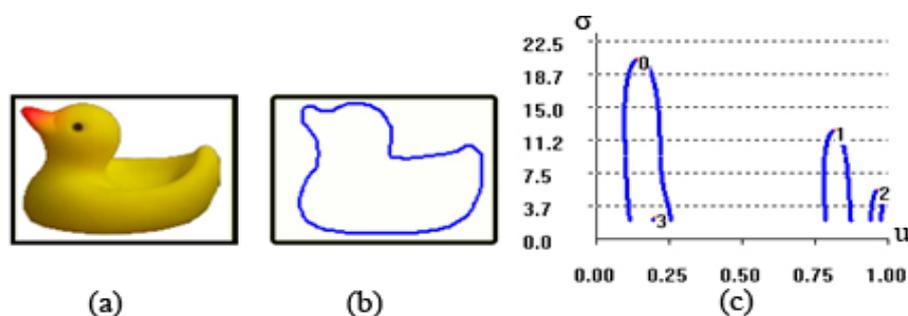
### 2.2. Methods with a Dynamic Number of Views

To overcome those limitations of view under and over views estimation due to the fixed 2D view number, a set of methods has been proposed for automatic selection of the optimal views of an object: these methods eliminate similar views and select a relatively small number of views using an optimization algorithm. This number varies depending on the complexity of the object and the measure of expected accuracy.

Ansary *et al.* [18] propose an algorithm, called adaptive view clustering to choose the characteristic views of a 3D model. Their method relates the number of views to its geometrical complexity. Starting from 320 viewpoints, equally spaced on the bounding sphere, the algorithm selects the optimal views, clustering them with the Zernike moment descriptor. The resulting number of views varies from 1 to 40, depending on the object complexity.

Mokhtarian and Abbasi [19] propose a method that eliminates the similar views in the sense of a distance among curvature scale space (CSS) from the outlines of these views. In the end, the minimal number of views is selected with an optimization algorithm. The CSS image of a planar curve [20] is computed by convolving a path-based parametric representation of the curve with a Gaussian function of increasing variance  $\sigma^2$ , extracting the zeros of curvature of the convolved curves and combining them in a scale space representation for the curve. The result of the mapping is usually an interval tree, called

a CSS map, consisting of inflection points. The peaks of the CSS contour map are extracted out and sorted in descending order. Figure 1 shows the obtained CSS map for a given view in the COIL-100 database. Small contours of the binary image are related to noise or small ripples of the curve. As a result, small maxima are not included in the CSS representation. Finally, the descriptor is presented as:  $\{(20.40, 0.149); (12.20, 0.82); (5.50, 0.97); (2.30, 0.21)\}$ .



**Figure 1.** (a) Example of view from COIL-100 database; (b) its contour; (c) and its curvature scale space (CSS) image in the  $(u, \sigma)$  plane.

For all of the 2D/3D approaches presented, two major problems are encountered: how to characterize the 3D model with a small number of 2D views; and how to use these views to find the model of a collection of 3D models. In the next section, we present a new method for the optimal selection of 2D views from a 3D model based on the pivot selection techniques for proximity searching in metric spaces.

### 3. Proposed Method

The main idea of the 2D/3D approach is to represent a 3D model by a set of characteristic views. The 3D object is scaled with respect to its bounding sphere, and principal component analysis (PCA) is applied in order to normalize the pose of the model by estimating the principal axes of a 3D object that are used to determine its orientation [21].

Let  $V$  be the space of views of the 3D model discretized to  $M$  points of the view distributed around the 3D model and  $d$  the distance metric of the space. In order to select an optimum view, we propose to use the pivot selection techniques for the proximity search in metric spaces. The set of pivots selected from space  $V$  of views will present the optimum view of the 3D object treated. We propose to select optimum views by using the incremental algorithm described in [8]; it is one of the most suitable strategies for real-world metric spaces [22], in which the authors suggest an efficiency criterion that compares two set of pivots and designates the better of the two.

Let  $\{v_1, v_2, \dots, v_k\}$  be a set of  $k$  pivots, with  $v_i \in V$ .  $\{v_1, v_2, \dots, v_k\}$  defining a space  $P$  of distance tuples between pivots and views from  $V$ . The mapping of a view  $u \in V$  to  $P$ , which will be denoted  $[u]$ , is carried out as  $[u] = (d(u, v_1), d(u, v_2), \dots, d(u, v_k))$ . Defining the metric  $D_{\{v_1, v_2, \dots, v_k\}}([x], [y]) = \max_{\{1 \leq i \leq k\}} |d(x, v_i) - d(y, v_i)|$ , it follows that  $(P, D)$  is a metric space, which turns out to be  $(R^k, L_\infty)$ .

Let  $\mu_V$  be the mean of the distribution  $V$ ; the authors in [8] suggest an efficiency criterion that compares two sets of pivots and designates the better of the two. Given two sets of pivots  $P_1 = \{p_1, p_2, \dots, p_k\}$  and  $P_2 = \{p_1', p_2', \dots, p_k'\}$ , we call  $P_1$  better than  $P_2$  if  $\mu_{\{p_1, p_2, \dots, p_k\}} > \mu_{\{p_1', p_2', \dots, p_k'\}}$ .

An estimation of the value of  $\mu_V$  is obtained as follows:  $A$  pairs of views  $\{(a_1, a'_1), (a_2, a'_2), \dots, (a_A, a'_A)\}$  from  $V$  are chosen at random. All of the pairs of views are mapped to space  $P$ , obtaining the set  $\{D_1, D_2, \dots, D_A\}$  of distances  $D$  between every pair of views. The value of  $\mu_V$  is estimated as  $\mu_V = \frac{1}{A} \sum_{1 \leq i \leq A} D_i$ .

The processes of selecting the optimum view of 3D object works as follows: the first view  $v_1$  is selected from a sample of  $N$  views of  $V$ , such that the view alone has the maximum  $\mu_D$  value. Then, a second view  $v_2$  is chosen from another sample of  $N$  views of  $V$ , such that  $\{v_1, v_2\}$  has the maximum  $\mu_D$  value, considering  $v_1$  as fixed. The third view  $v_3$  is chosen from another sample of  $N$  views of  $V$ , such that  $\{v_1, v_2, v_3\}$  has the maximum  $\mu_D$  value, considering  $v_1$  and  $v_2$  as fixed. The process is repeated until  $k$  views have been chosen. The set of views  $\{v_1, v_2, \dots, v_k\}$  selected from  $V$  will be the optimum set of views of the 3D object. For each iteration  $i$  of the algorithm, we propose to select the sample  $N$  views from the  $i^{\text{eme}}$  cluster  $c_i$  of views of the 3D object obtained by applying the  $k$ -means algorithm. The  $N$  views selected include the center of cluster  $c_i$ .

Let *EvaluationSetA* be a function that returns the set of  $A$  pairs of views from  $V$  and *CandidatePivot* be a function that returns the sample  $N$  views, including the center of cluster  $c_i$ . The steps of the proposed algorithm are illustrated in Algorithm 1.

---

**Algorithm 1.** The proposed algorithm.

---

**Input:**

$V$ : the set of views of 3D object  
 $d$ : the distance metric used to compare two views  
 $NP$ : the number of optimum views

**Output:**

*SetPivot*: set of optimum views

**Variables:**

*setA*: the set of  $A$  pairs of views used to estimates  $\mu_V$   
*setC*: the sample  $N$  views, including the center of cluster  $c_i$  //  $setC \subset cluster_i$

**Begin**

Classify the views of  $V$  into  $NP$  clusters using a  $k$ -means clustering algorithm

$setA = EvaluationSetA(V, d)$ ;

$SetPivot = \emptyset$ ;

for  $i = 1$  to  $NP$  do //  $NP$  present the number of clusters

$setC = CandidatePivot(cluster_i, d)$ ; //  $c_i \in setC$

$bestValue = 0$ ;

for (each  $x$  in  $setC$ )

$value = GetValueUv(setE, d, P \cup x)$ ;

if ( $value$  is better than  $bestValue$ )

$bestValue = value$ ;

$bestView = x$ ;

endif

endif

$SetPivot = SetPivot \cup bestView$ ;

endfor

Return  $SetPivot$ ;

**End**

---

The function *GetValueUv* used to estimate the value of  $\mu_V$  exploits a mapping of the form  $\psi: (V, d) \mapsto (\mathbb{R}^p, L_\infty)$ , defined by  $k$  pivots  $v_i$ , such that  $\psi(x, y) = L_\infty(\psi(x), \psi(y)) = \max_{\{1 \leq i \leq k\}} |d(x, v_i) - d(y, v_i)|$ , where  $\psi(x) = [d(P_1, x), d(P_2, x), \dots, d(P_p, x)]$ . The steps of *GetValueUv* are illustrated in Algorithm 2.

---

**Algorithm 2.** The function GetValueUv.

---

**Input:**

*setE*: the set of  $A$  pairs of views  
*d*: the distance metric used to compare two views  
*setP*: the set of pivots  $\{v_1, v_2, \dots, v_k\}$

**Output:**

$\mu_V$ : the mean of the distribution

**Begin**

- all pairs of *setE* are mapped into the vector space associated with the set of pivots *setP* using the mapping function  $\psi(\cdot)$
- for every pair  $(x, y)$ , compute the distance between  $x$  and  $y$  in the feature space, that is  $d_i = L_\infty(\psi(x), \psi(y)) = \max_{\{1 \leq i \leq k\}} |d(x, v_i) - d(y, v_i)|$
- compute  $\mu_V$  as the mean of these distances, *i.e.*,  $\mu_V = \frac{1}{A} \sum_{1 \leq i \leq A} d_i$

**End**


---

If the distances  $D_{\{p_1, \dots, p_{i-1}\}}([a_r], [a'_r])$ ,  $1 \leq r \leq A$ , are kept in an array, it is not necessary to redo all of the distance computations to estimate  $\mu_V$  when the  $i$ -th pivot is added. It is enough to calculate  $D_{\{p_i\}}([a_r], [a'_r])$ ,  $1 \leq r \leq A$ , because it follows that  $D_{\{p_1, \dots, p_i\}}([a_r], [a'_r]) = \max(D_{\{p_1, \dots, p_{i-1}\}}([a_r], [a'_r]), D_{\{p_i\}}([a_r], [a'_r]))$ . Therefore, only  $2NA$  distance computations are needed to estimate  $\mu_V$  when a new pivot is added, where  $N$  presents the number of optimum views and  $A$  presents the number of pairs of views. Since the process is repeated  $k$  times, the total cost is  $2kAN$  for distance computations.

We have indexed each view by using four well-established descriptors from the MPEG-7 standard [23,24] that captures various image characteristics.

The scalable color descriptor (SCD) is derived from a color histogram in the hue-saturation-value color space with fixed space quantization. We used the 64 coefficients version of this descriptor.

The color structure descriptor (CSD) aims at identifying localized color distributions using an  $8 \times 8$  pixel structuring matrix that slides over the image. This descriptor can distinguish between two images having a similar amount of pixels of a specific color, if structures of these pixels differ in these images.

The color layout descriptor (CLD) is obtained by applying the discrete cosine transform on a 2D array of local representative colors in three color channels.

The edge histogram descriptor (EHD) represents the local-edge distribution in the image. The images are subdivided into  $4 \times 4$  sub-images, and edges in each sub-image are categorized into five types: vertical, horizontal,  $45^\circ$  diagonal,  $135^\circ$  diagonal and non-directional edges. This results in 80 coefficients representing the local edge histograms. Furthermore, these semi-global and the global histograms can be computed based on the local histogram.

#### 4. Measure of Similarity

Distance represents a way of quantifying the closeness of objects in metric space. In order to measure the similarity between two 3D objects, we propose to use a Hausdorff distance, one of the commonly-used measures for object matching. Let  $X = \{\bar{x}_i / 1 \leq i \leq n\}$  be a set of optimum views of an input 3D object, where  $n$  is the number of views. Additionally, let  $Y = \{\bar{y}_j / 1 \leq j \leq m\}$  be a set of views of a stored object 3D, where  $m$  is the number of views.

The distance between a view  $x_i$  of 3D object  $X$  and 3D object  $Y$  is defined as:

$$d(x_i, Y) = \min_{y_j \in Y} d(x_i, y_j) \quad (1)$$

where  $d(x_i, y_j)$  is the distance between views of 3D object  $X$  and 3D object  $Y$ .

The formula of the Hausdorff distance used to measure the similarity between 3D objects  $X, Y$  is:

$$d_H(X, Y) = \max(d_E(X, Y), d_E(Y, X)) \quad (2)$$

where:

$$D_E(X, Y) = \sup_{x_i \in X} d(x_i, Y) \quad (3)$$

Each view is described by four MPEG-7 well-established descriptors from the MPEG-7 standard [23,24]. Therefore, the distance used in the proposed method to calculate the distance between two views  $x_i$  and  $y_j$  is defined as follows:

$$d(x_i, y_j) = \alpha_1 D\_CSD(x_i, y_j) + \alpha_2 D\_SCD(x_i, y_j) + \alpha_3 D\_EHD(x_i, y_j) + \alpha_4 D\_CLD(x_i, y_j) \quad (4)$$

where:

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 1/4$$

and:

$D\_SCD$  is the  $L_1$  metric used to compare two scalable colors.

$D\_CSD$  is the  $L_1$  metric used to compute color structure descriptor distances.

$D\_CLD$  is the distance between two views computed as a sum of  $L_2$  distances in each of the three color space components.

$D\_EHD$  is computed as a sum of weighted sub-sums of absolute differences for the local, semi-global and global histograms.

## 5. Experimental Results

We have implemented a system for the indexing and retrieving of 2D/3D models. The application has been developed in visual C++. To test the reliability of the proposed method, we use two databases, COIL-100 and The Amsterdam Library of Images (ALOI-1000).

### 5.1. COIL-100 Database

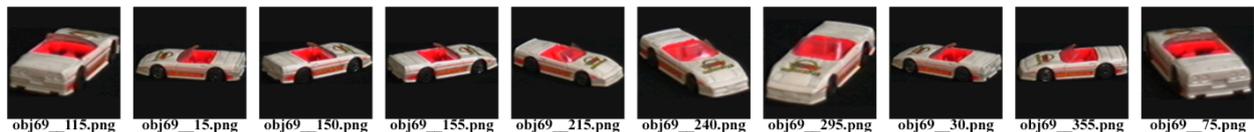
The Columbia Object Image Library [25] consists of 100 objects. Each object was placed on a motorized turntable, which was rotated through 360 degrees with respect to a fixed camera. Images of the objects were taken at pose intervals of  $5^\circ$ . This corresponds to 72 images per object, making a total of 7200 gray-scale images. The objects have a wide variety of complex geometric and reflectance characteristics. The images are a normalized size. In Figure 2, the frontal pose of each object is shown.



**Figure 2.** Samples of images from the COIL-100 image database.

We have used the COIL-100 database because it is known as a very standard object image database, and it has several views of the same object taken at different poses. We have described each view by four descriptors, *SCD*, *CSD*, *CLD* and *EHD*. The distance used in the proposed method to compare views is described in Section 4.

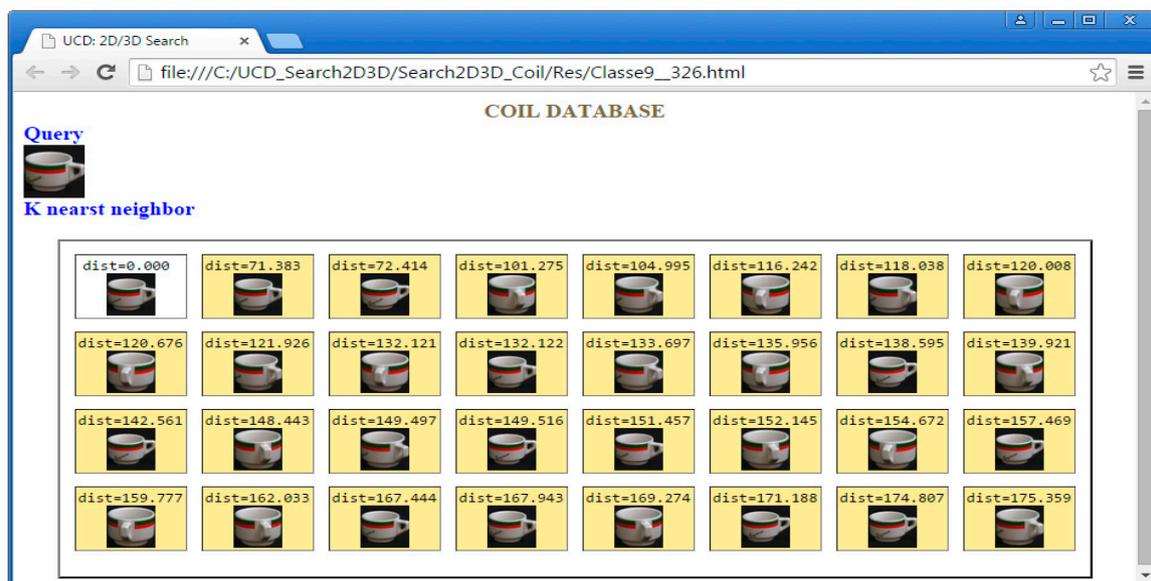
Figure 3 presents the 10 optimum views of Class 69 of the COIL-100 database obtained by applying the proposed method.



**Figure 3.** The 10 selected views of Class 69 of the COIL-100 database.

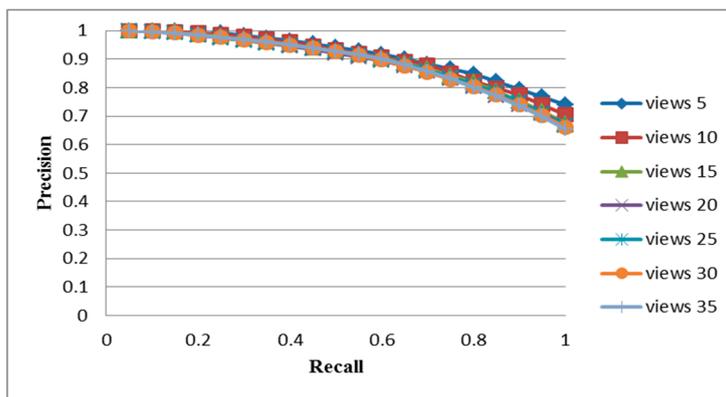
Views already existing in the COIL-100 database can be used as a query by the user; then, we have developed a program by exploiting some tools provided by Princeton Shape Benchmark [26], for generating web pages to see results of each query. The output is a series of .html pages; each page contains the views similar to the user query from the database in decreasing order of similarity using the *k*-nearest neighbor algorithm.

An example of query view in the COIL-100 database is shown in Figure 4. The obtained results are similar to those that a user could find visually.



**Figure 4.** Example of a query and its 32 similar views obtained by using the *k*-nearest neighbor algorithm.

The retrieval performances of our methods are assessed by using the average precision-recall curves. Precision *P* is defined as the ratio of the number of retrieved relevant views to the total number of retrieved views *n*, *i.e.*,  $P = r/n$ . Recall *R* is defined as the ratio of the number of retrieved relevant views *r* to the total number *m* of relevant views in the whole database, *i.e.*,  $R = r/m$ . Results in terms of interpolated precision vs. recall curves for different values of number of views are shown in Figure 5.

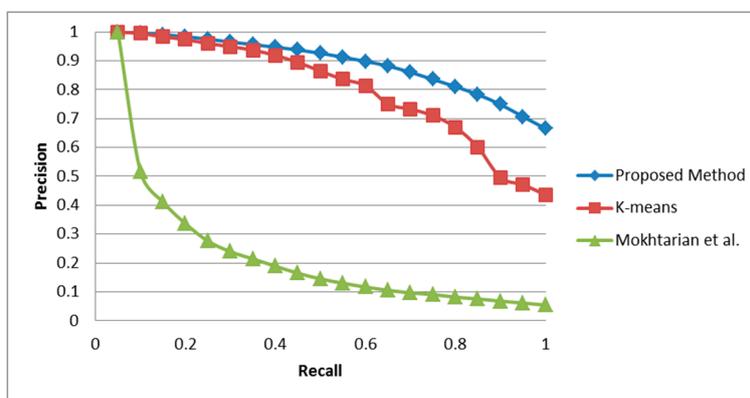


**Figure 5.** Average precision-recall curves of the proposed method for different values of views.

It can be observed that the proposed method for different values of the number of selected views almost always presents good performance. Indeed, precision values decrease weakly, and all curves have precision beyond 60%. The best results are obtained in the case of the number of optimum views equal to five.

To check the importance of our application, we have compared the proposed method to two methods. The first one is *k*-means clustering, one of the classical, well-studied unsupervised learning algorithms that solve the fundamental clustering problem [27] and also one of the known methods employed in 3D retrieval [17,18,28,29]. To calculate optimum views by using *k*-means, first, randomly choosing *k* views as initial cluster centers from views of 3D object, the remaining part of each view, according to its distance with various cluster centers, gets its recent or most similar clustering respectively; and then, recalculate the average value of each cluster as a new cluster, and adjust the various types of samples; this process is repeated until all of the samples are in their sub-cluster with the minimum distance square. The center of clusters presents the optimum views of the 3D object.

The second method was introduced by Mokhtarian *et al.* [19] and is based on an automatic selection of optimal views. The shapes of views are described by the CSS descriptor, which was one of the features selected to describe 2D objects in the MPEG-7 standard [30]. The maxima of the CSS image are used to represent two-dimensional shapes at different levels of resolution. The CSS descriptor is employed to select the optimal views by matching the rendered views and discarding the similar views whose matching costs fall in a predefined threshold.



**Figure 6.** Average precision-recall curves of the proposed method, *k*-means and the automatic selection of optimal views proposed by Mokhtarian *et al.* [19].

The performances of the proposed method,  $k$ -means and automatic selection of optimal views based on CSS [19] are measured in the case of the number of optimum of views equal to 25 by using recall-precision curves. The obtained results are shown in Figure 6.

It can be seen from the precision-recall curves that the proposed descriptor outperforms the automatic selection method-based CSS and  $k$ -means method. These results prove the performance of the proposed method and the importance to integrate the pivot-based algorithm in 2D/3D search.

The use of efficient criteria [8] to select the  $i^{\text{eme}}$  optimum view from a sample views  $N$  of the  $i^{\text{eme}}$  cluster, in which center of cluster is part of  $N$ , instead of selecting the center of the cluster as an optimum view in  $k$ -means, explains the better results of our method than the  $k$ -means method.

Automatic selection of optimal views [19] only uses a shape descriptor CSS to describe views. However, many views of 3D object contains color and texture information, so the performance of the automatic selection of optimal view-based CSS is significantly lower than the proposed method and the  $k$ -means method.

The experiments on the COIL-100 database confirm that our method has higher precision at all recall levels. In order to prove the performance of the proposed method in a big database, we have used ALOI-1000 with 72,000 views of color images.

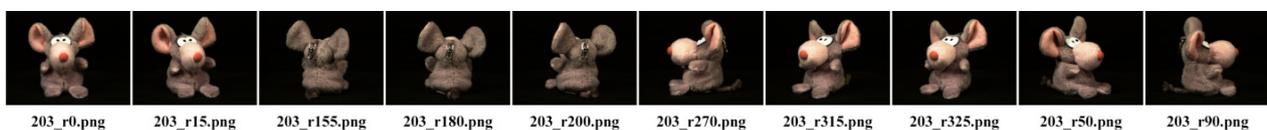
## 5.2. ALOI-1000 Database

Amsterdam Library of Images ALOI-1000 [31] consists of 1000 views with 72 color images of views from zero to 360 degrees taken in five-degree steps. In Figure 7, samples of images from the ALOI database are shown.



**Figure 7.** Samples of images from the Amsterdam Library of Images (ALOI) database.

Figure 8 shows the 10 optimum views of Class 203 of the ALOI-1000 database obtained by applying the proposed method.

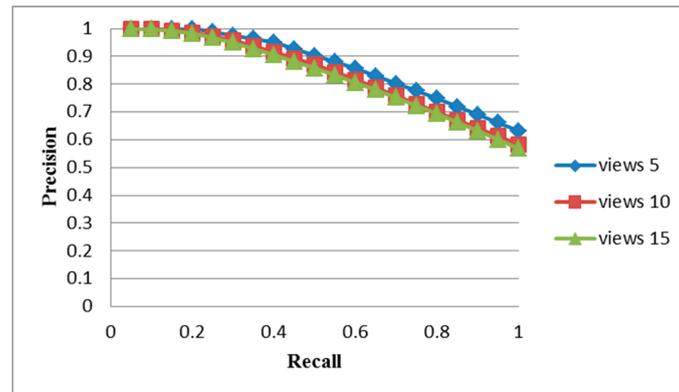


**Figure 8.** The 10 selected views of Class 203 of the ALOI-1000 database.

In order to measure the performance of the proposed method in the ALOI-1000 database, we have calculated the average recall/precision curves for three values of the number of optimum views, 5, 10 and 15. The obtained results are shown in Figure 9.

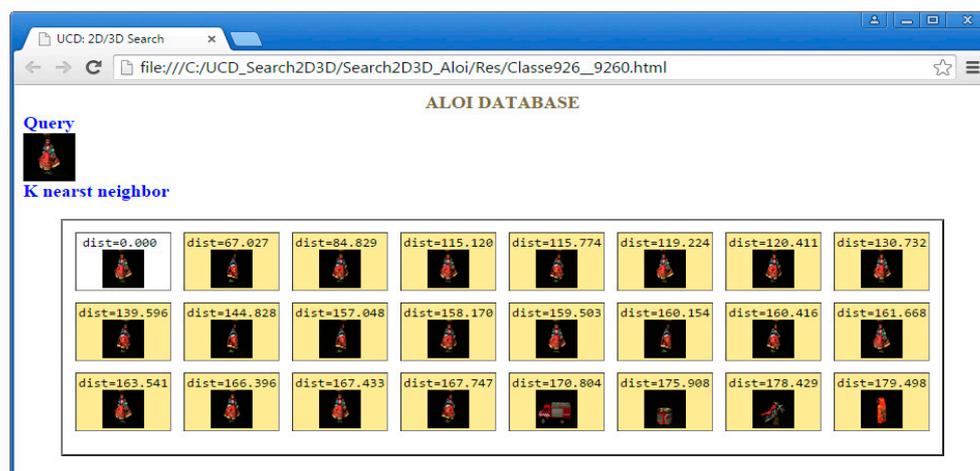
It can be seen from the precision-recall charts that the proposed method presents a good performance. These results prove the performance of the proposed method and the efficiency of the descriptors used to describe each view in the COIL-100 and ALOI-1000 databases. The best results are obtained in the case of number of optimum views equal to five.

We have used the views of the ALOI-1000 database as a query; the obtained results are displayed in decreasing order of similarity by using the  $k$ -nearest neighbor algorithm.



**Figure 9.** Precision-recall curves of the proposed method for three numbers of optimum views, 5, 10 and 15.

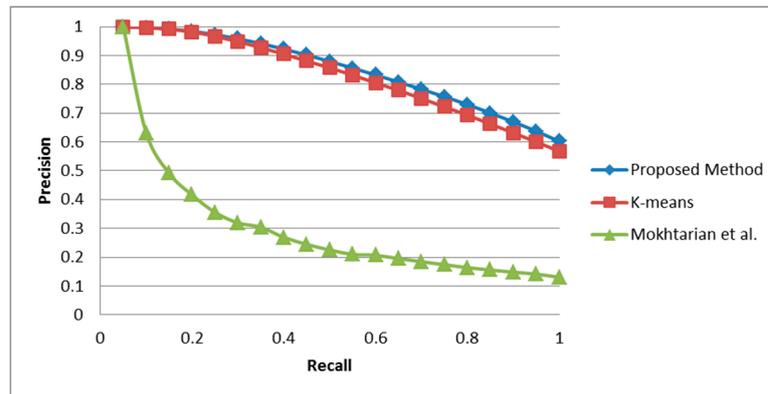
An example of query is shown in Figure 10. The obtained results are similar to those that a user could find visually.



**Figure 10.** Example of a query and its 25 similar views obtained by using the  $k$ -nearest neighbor algorithm.

We have compared the performance of the proposed method with the  $k$ -means and automatic selection of optimal views proposed based on CSS [19] in the case that the number of pivots is equal to 15. The result of the curve recall/precision is presented in Figure 11.

It can be seen from the precision-recall charts that the proposed method outperforms  $k$ -means and the automatic selection of optimal views proposed by Mokhtarian *et al.* [19]. These results prove the performance of the proposed method and the importance of using the pivot selection techniques in metric space to select the optimum views of 3D objects.



**Figure 11.** Average precision-recall curves of the proposed method, *k*-means and the automatic selection of optimal views proposed by Mokhtarian *et al.* [19].

## 6. Conclusions and Future Work

In this paper, a new method for selecting optimum views has been proposed. It is based on the pivot selection techniques for proximity searching in metric spaces. The selection of optimum views is obtained by using an incremental algorithm based on an efficiency criterion that compares two pivot sets and designates the better of the two. The selected views have been described by four descriptors, CSD, SCD, CLD and EHD, from the MPEG-7 standard. The method was tested on the ALOI database and the COIL database by using a combination of metrics to measure the similarity between views and recall/precision to measure the performance of the search in the database for different values  $k$ , where  $k$  represents the number of optimum views. The result obtained is very encouraging and shows the interest of our method and its superiority over *k*-means and the automatic selection of optimal views based on curvature scale space in the ALOI-1000 and COIL-100 databases. In future work, we plan to apply the proposed method to other related applications, like face recognition, video and search by sketch.

## Author Contributions

Both authors are responsible for the concept of the paper, the results presented and the writing. Both authors have read and approved the final manuscript.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Silkan, H.; Tmiri, A.; El Alaoui Ouatik, S.; Lachkar, A. A new shape descriptor 2D for Content Based Image Retrieval. In Proceedings of the 2014 Second World Conference Complex Systems (WCCS), Agadir, Morocco, 10–12 November 2014; pp. 670–674.
2. Haythem, B.; Mohamed, H.; Marwa, C.; Fatma, S.A. Fast Generalized Fourier Descriptor for object recognition of image using CUDA. In Proceedings of the 2014 World Symposium on Computer Applications & Research (WSCAR), Sousse, Tunisia, 18–20 January 2014; pp. 1–5.

3. Qiuying, Y.; Ying, W. Zernike moments descriptor matching based symmetric optical flow for motion estimation and image registration. In Proceedings of the 2014 International Joint Conference on Neural Networks (IJCNN), Beijing, China, 6–11 July 2014; pp. 350–357.
4. Silkan, H.; El Alaoui Ouatik, S.; Lachkar, A.; Mekkassi, M. A Novel Shape Descriptor Based on Extreme Curvature Scale Space Map Approach for Efficient Shape Similarity Retrieval. In Proceedings of the 2009 Fifth International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), Marrakesh, Morocco, 29 November–4 December 2009; pp. 160–163.
5. Hanyf, Y.; Silkan, H.; Labani, H. Criteria and technique for choice good  $\rho$  value for D-index. In Proceedings of the International conference on Intelligent Systems and Computer Vision, Fez, Morocco, 25–26 March 2015.
6. Chen, L.; Gao, Y.; Li, X.; Jensen, C.S.; Chen, G. Efficient metric indexing for similarity search. In Proceedings of the 2015 IEEE 31st International Conference on Data Engineering (ICDE), Seoul, Korea, 13–17 April 2015.
7. Tosun, U. A novel indexing scheme for similarity search in metric spaces. *Pattern Recognit. Lett.* **2015**, *54*, 69–74.
8. Bustos, B.; Navarro, G.; Chavez, E. Pivot selection techniques for proximity searching in metric spaces. In Proceedings of the International Conference of the Chilean Computer Science Society, Punta Arenas, Chile, 7–9 November 2001; pp. 33–40.
9. Mico, L.; Oncina, J.; Vidal, R. A new version of the nearest neighbor approximating and eliminating search (AESAs) with linear pre-processing time and memory requirements. *Pattern Recognit. Lett.* **1994**, *15*, 9–17.
10. Yianilos, P. Data structures and algorithms for nearest-neighbor search in general metric spaces. In Proceedings of the 4th ACM-SIAM Symposium on Discrete Algorithms, Austin, TX, USA, 25–27 January 1993; pp. 311–321.
11. Brin, S. Near neighbor search in large metric spaces. In Proceedings of the 21st Conference on Very Large Databases, Zurich, Switzerland, 11–15 September 1995; pp. 11–15.
12. Karina, F.; Rodrigo, P. An Effective Permutant Selection Heuristic for Proximity Searching in Metric Spaces. In Proceedings of the 6th Mexican Conference, MCPDR 2014, Cancun, Mexico, 25–28 June 2014; pp. 102–111.
13. Chen, D.; Tian, X.; Shen, Y.; Ouhyoung, M. On visual similarity based 3D model retrieval. *Comput. Graph. Forum* **2003**, *22*, 223–232.
14. Kazhdan, M.; Funkhouser, A.S. Rusinkiewicz, Rotation invariant spherical harmonic representation of 3D shape descriptors. In Proceedings of the Eurographics Symposium on Geometry Processing, Aachen, Germany, 23–25 June 2003; pp. 156–164.
15. Funkhouser, T.; Min, P.; Kazhdan, M.; Chen, J.; Halderman, A.; Dobkin, D.; Jacobs, D. A Search Engine for 3D Models. *ACM Trans. Graph.* **2003**, *22*, 83–105.
16. Chaouch, M.; Verroust-Blondet, A. A new descriptor for 2D depth image indexing and 3D model retrieval. In Proceedings of the IEEE International Conference on Image Processing, ICIP 2007, San Antonio, TX, USA, 16 September–19 October 2007; pp. 373–376.
17. Ramezani, M.; Ebrahimnezhad, H. 3D Models' Retrieval System Design Based on Poisson's Histogram of 2D Selective Views. In Proceedings of the 21st Iranian Conference on Electrical Engineering (ICEE), Mashhad, Iran, 14–16 May 2013.

18. Ansary, T.F.; Daoudi, M.; Vandeborre, J.P. A bayesian 3D search engine using adaptive views clustering. *IEEE Trans. Multimed.* **2007**, *9*, 78–88.
19. Mokhtarian, F.; Sadegh, A. Robust automatic selection of optimal views in multi-view free-form object recognition. *Pattern Recognit.* **2005**, *38*, 1021–1031.
20. Mokhtarian, F.; Bober, M. *Curvature Scale Space Representation: Theory Application and MPEG-7 Standardization*; Springer: Berlin, Germany, 2003.
21. Konstantinos, S.; Theoharis, T.; Ioannis, P. ROSy+: 3D Object Pose Normalization Based on PCA and Reflective Object Symmetry with Application in 3D Object Retrieval. *Int. J. Comput. Vis.* **2011**, *91*, 262–279.
22. Zezula, P.; Amato, G.; Dohnal, V.; Batko, M. *Similarity Search: The Metric Space Approach (Advances in Database Systems)*; Springer: Berlin, Germany, 2006.
23. Manjunath, B.S.; Salembier, P.; Sikora, T. (Eds.) *Introduction to MPEG-7: Multimedia Content Description Interface*; John Wiley & Sons, Inc.: New York, NY, USA, 2002.
24. Jakub, L.; Novák, D.; Batko, M.; Skopal, T. Visual Image Search: Feature Signatures or/and global Descriptors, similarity search and applications. In Proceedings of the 5th International Conference, SISAP 2012 Toronto, ON, Canada, 9–10 August 2012.
25. Nene, S.A.; Nayar, S.K.; Murase, H. *Columbia Object Image Library (COIL-100)*; Technical Report CUCS-006-96; Columbia University: New York, NY, USA, 1996.
26. Shilane, P.; Min, P.; Kazhdan, M.; Funkhouser, T. The Princeton Shape Benchmark. In Proceedings of the International Conference on Shape Modeling and Applications (SMI'04), Genoa, Italy, 7–9 June 2004; pp. 167–178.
27. MacQueen, J.B. Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, CA, USA, 21 June–18 July 1965; pp. 281–297.
28. Kao, C.-C. View Based Rotation Invariant 3D Shape Retrieval. In Proceedings of the International Conference on Remote Sensing and Data (ICRSD), Hong Kong, 2–3 July 2011; pp. 150–155.
29. Mathias, E.; Hildebrand, K.; Boubekur, T.; Alexa, M. Sketch-based 3D shape retrieval. In Proceedings of the SIGGRAPH 2010 Talks, Los Angeles, CA, USA, 25–29 July 2010.
30. Bober, M. MPEG-7 Visual Shape Descriptors. *IEEE Trans. Circuits Syst. Video Technol.* **2001**, *11*, 716–719.
31. Geusebroek, J.M.; Burghouts, G.J.; Smeulders, A.W.M. The Amsterdam library of object images. *Int. J. Comput. Vis.* **2005**, *61*, 103–112.