

# ThOR: Three-dimensional Object Retrieval Library

Pedro B. Pascoal<sup>1,2</sup> and Alfredo Ferreira<sup>1</sup>

<sup>1</sup>INESC-ID / Instituto Superior Técnico / Universidade de Lisboa

<sup>2</sup>Microsoft Language and Development Center, Lisbon, Portugal

pmbp@tecnico.ulisboa.pt, alfredo.ferreira@tecnico.ulisboa.pt

---

## Abstract

Following the increasing number of 3D object collections, researchers developed several algorithms related to 3D object analysis, comparison and retrieval methods. However, there is no simple solution offering researchers and practitioners a framework for the integration of algorithms and techniques developed within this context into their applications and tools.

ThOR (Three-dimensional Object Retrieval) is a lightweight open source Java library for content based object retrieval, that provides common 3D shape retrieval indexing and retrieval tools. Most important, it allows addition of new components, such as shape descriptors, with minimal effort. In short, ThOR provides an easy solution for the implementation of 3D object retrieval tools using both local and internet-based client-server architectures.

Categories and Subject Descriptors (according to ACM CCS): H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—

---

## 1. Introduction

The increasing number of three-dimensional objects available on digital format, has triggered a great interest in finding efficient methods of analysis, comparison and retrieval of 3D models. However, despite the existence of benchmarks with collections of 3D models [SMKF04, MR08], annual contests with specific tracks to compare techniques [MR08, MFP\*13, LLL\*14], and even a quite complete repository of shapes and tools [AIM11], there is no integrated library that provides the necessary tools to easily incorporate the various techniques associated with 3D object retrieval into applications.

The work we carried out focuses on the development of a simple and centralized library for 3D shape analysis, classification and retrieval, that allows researchers and practitioners to use, compare and combine techniques and tools. Furthermore, our solution also enables researchers to easily add their algorithms to the library, further expanding the number of available tools for 3D object analysis and retrieval.

Notably, a few years ago, Lux and Chatzichristofis proposed a library for content-based image retrieval (CBIR) [LC08]. LIRe (Lucene Image Retrieval) is an open source library that providing multiple common and state of the art retrieval mechanisms for image analysis.

This library, is actively used for research, teaching and commercial applications, and clearly demonstrates the many benefits, of having a centralized library with state of the art retrieval mechanisms. Developers and researchers can easily extend and modify LIRe to adapt it to their needs.

LIRe served as a catalyst for the development of our library for the 3D object retrieval domain. While still a work-in-progress, ThOR is available for download<sup>†</sup>.

In next section we present an overview of presented library, briefly describing its main features. Next, in Section 3, we describe some of technical details and architecture. Finally, we present a demo application created using the ThOR library, followed by some conclusions and point out future work.

## 2. THOR

ThOR is an open source Java library for content based 3D object analysis, classification and retrieval. ThOR provides a framework to easily incorporate and share tools and techniques in this context into applications. Indeed, to the extend of our knowledge, there is no unified solution that provides

---

<sup>†</sup> <http://3dorus.ist.utl.pt/tools/thor.html>

an integrated framework that offers multiple tools for analysis and retrieval of 3D objects.

Similarly to LIRe in image retrieval domain, ThOR offer developers and researchers a simple framework for 3D object retrieval. As a starting point, following the classic approach to retrieval, we developed the ThOR library with three main objectives in mind:

- **Ease-to-use:** provide a basic set of techniques that would enable the creation of 3DOR applications using minor effort;
- **Scalable:** allows extension of ThOR, easily adding new techniques and tools;
- **Open:** offers integration with other libraries, such as LIRe and Lucene.

Furthermore, by providing a library to aggregate methods for 3DOR, ThOR offers researchers with the means to compare classic 3DOR approaches with their newer techniques. Also, since we made it scalable, newer tools can easily be integrated to further extend the functionality of ThOR. ThOR is available online, licensed under GPL.

### 3. Technical Details

ThOR was built based on the concepts behind LIRe library, but it is an independent framework. Nevertheless, some prototypes and demos available with ThOR contain some dependencies, although the main library of ThOR itself is completely self-sufficient. In this section we provide an overview of the main features available with this retrieval library, describing some of the different low level features that are available.

#### 3.1. Features

To provide basic functionality, we chose some existing algorithms and techniques, from different classification branches [TV08] that have proven to be useful and efficient. Among the current features, we would like to highlight the following:

- **Principal Component Analysis [PR99]**  
The most prominent tool for accomplishing the rotation invariance is Principal Component Analysis (PCA). The PCA algorithm assign the principal axes and attends a normalization of the rotation, using the eigenvalues.
- **Shape Distributions [OFCD02]**  
Although shape distributions are rarely used as full system model descriptors, given their not so high average retrieval rates, they are commonly used as pre-classifiers because they also tend to be somewhat stable across different databases, and used in conjunction with other descriptors.
- **Coord and Angle Histograms [PR99]**  
A cord is defined as a ray segment which joins the barycenter of the mesh with a triangle center. Since only

global features are used to characterize the overall shape of objects this method is not very discriminative about object details, but the implementation is straightforward. Like Shape Distributions, these methods can be used as an active filter or in combination with other methods to improve results.

- **Skeleton graphs [SSG\*03]**  
The skeletal graphs provides the means to represent the object according to their topological similarity. The topological information is created using the approach proposed by Sundar et al. [SSG\*03] which a skeletal graph is built and used graph matching techniques to match the skeletons to compare 3D objects.
- **Lightfield Descriptors [CTSO03]**  
The Light Field Descriptor is a view-based geometry feature extractor. Its authors claim, for studies driven in different databases, that its retrieval rate is distinctively higher than other view-based and feature-based methods, and that it should be tested with other benchmarks [SMKF04].
- **2D View Extraction**  
Using partially the approach of the Light Field descriptors, we provide a set of features for the extraction of views of the objects. These view can be as silhouettes, contours or sketch-based, providing the means to compare 3D objects with 2D image.

Beside the described features, we also provide basic techniques for object transformation, and calculation of 3D object basic features, such as barycenter, convex hull, surface area, and so on. Additionally, users and developers can easily add their own algorithms and techniques, sharing them with the community or not.

#### 3.2. Indexing

While most features stated above focus specifically in the analysis and retrieval of three-dimensional objects, the indexing mechanism is an essential component of any information retrieval system. ThOR includes such component amongst its basic features. To accelerate the searching of specific data items most database systems use a tree indexing, reducing the number of elements to be compared. Nevertheless, tree-based techniques do not seem to scale up well with growing data set sizes, typical of multimedia databases, such as 3D object collections. Indeed, it is relatively easy to assemble large collections of objects, whereas the overhead introduced by tree management far outweighs the potential advantages of sophisticated spatial indexing structures.

A solution for this problem, is the NB-Tree proposed by Fonseca and Jorge [FJ03]. The NB-Tree aims to offer a compact and simple solution for indexing large volumes of multimedia data. In a first step, it maps multidimensional points to a 1D line by computing their Euclidean Norm. Then, in a second step, it sort these points using a B<sup>+</sup>-Tree on which all subsequent operations are performed, as illustrated in Fig-

ure 1. This will help reduce the cost of sequential scan of the elements of  $B^+$ -Tree.

Furthermore, evaluation results show that it not only outperforms other methods, whereas the difference increases as the data dimensionality and data set size increase. As such, it shows to be able to handle data of variable dimensions in an efficient and scalable manner suitable for the format of feature vector used in objects comparison. Therefore we opt to integrate these simple, yet efficient, indexing mechanism into ThOR framework.

### 3.3. Retrieval

We exploit the NB-Tree indexing mechanism, which efficiently supports a wide type of queries, including point, range and the most used (at least in multimedia databases) nearest neighbor queries. When performing a query (of any type) the system examines points whose norm is in the neighborhood of the query point norm. Moreover, the  $B^+$ -Tree has the particularity that its leaves are linked as an ordered list. Thus, walking sequentially through all the elements of the  $B^+$ -Tree is a costless operation.

When querying, the system will only consider points whose norm is in the neighborhood of norm of the searched point, as illustrates Figure 1.

### 4. ThOR Web Demo

Using the described library as background we developed, with minimal effort, a simple web prototype to demonstrate the usage of ThOR. In overall, our prototype, enables users to explore and analyze a large 3D object collection, using a client-server architecture which can be broadly be subdivided into two parts: the backend and the frontend.

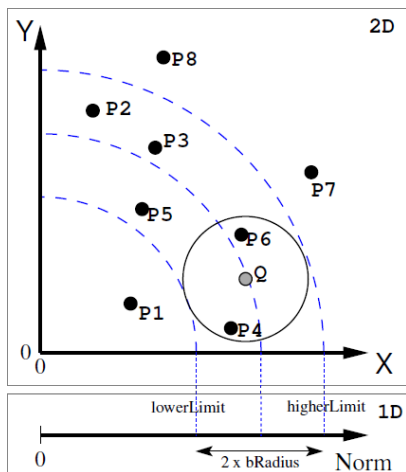


Figure 1: 2D Range query example [FJ03].



Figure 2: Demo result example.

The backend uses an offline pre-processed NB-Tree structure, built using features extracted from the Princeton Shape Benchmark [SMKF04]. The frontend, uses the previously described data to provide a REST-based API running on a cloud platform. For this we take advantage of the Play framework, which can easily be integrated with our library, due to also being a Java based framework. Thanks to this framework it's possible to query the prototype through the web interface or through requests contained as part of the URI. For this example we only focused on enabling simple query-by-example, using any of the 3DOR methods available in ThOR, as shown in Figure 2.

So basically, in this use-case, when submitting 3D object to the prototype, it is uploaded to the backend. Then, using the feature extraction and retrieval of ThOR, the k-NN most similar objects from the dataset are returned to the frontend.

While just an example of application of the ThOR framework, the above described example illustrates the simplicity of integrating the framework into a fully working solution.

### 5. Conclusions and Future Work

In this paper we presented ThOR, an open source Java library for content-based 3D object analysis, classification and retrieval. ThOR consists on a simple and centralized library that allows researchers and developers to use, study and compare available techniques and algorithms. Furthermore, due to its open source nature, it also enables researchers to easily add their own algorithms to the library, further extending ThOR by sharing them with the community. We designed our solution to be flexible and easily adaptable to different domains.

Nevertheless, the present solution is only an initial version of the framework. It is limited to the basic algorithms and techniques implemented. However, it can be easily extended, adding new functionalities and techniques to the already available ones. For example, the PCA sometimes erroneously assign the principal axes and produce inaccurate normalization results. Nevertheless, the addition of the rec-

tilinearity, proposed by Lian et al. [LRS10], combined with the PCA algorithm can minimize the normalization errors, providing a higher success of alignment. Additionally, the increase of the number of available 3DOR techniques, such as shape descriptors, would provide more variety for ThOR's usage in different scenarios.

## 6. Acknowledgements

This work was supported by national funds through "Fundação para a Ciência e a Tecnologia" (FCT) with reference UID/CEC/50021/2013, and partially supported by the "Agência Nacional de Inovação" (ANI) through the project CLUTCH, reference ADI/QREN/22984.

## References

- [AIM11] AIM@SHAPE: Shape repository, hosted at imati-cnr. <http://visionair.ge.imati.cnr.it/>, August 2011. [Online; accessed 13-April-2015]. 1
- [CTSO03] CHEN D.-Y., TIAN X.-P., SHEN Y.-T., OUHYOUNG M.: On Visual Similarity Based 3D Model Retrieval. vol. 22 of *EUROGRAPHICS 2003 Proceedings*, pp. 223–232. 2
- [FJ03] FONSECA M. J., JORGE J. A.: Nb-tree: An indexing structure for content-based retrieval in large databases. In *Proc. 8th Int. Conf. on Database Systems for Advanced Applications* (2003), pp. 267–274. 2, 3
- [LC08] LUX M., CHATZICHRISTOFIS S. A.: Lire: Lucene image retrieval: An extensible java cbir library. In *Proceedings of the 16th ACM International Conference on Multimedia* (New York, NY, USA, 2008), MM '08, ACM, pp. 1085–1088. URL: <http://doi.acm.org/10.1145/1459359.1459577>, doi:10.1145/1459359.1459577. 1
- [LLL\*14] LI B., LU Y., LI C., GODIL A., SCHRECK T., AONO M., BURTSCHER M., FU H., FURUYA T., JOHAN H., LIU J., OHBUCHI R., TATSUMA A., ZOU C.: Shrec'14 track : Extended large scale sketch-based 3d shape retrieval. In *Eurographics Workshop on 3D Object Retrieval, Strasbourg (France), 2014 : Proceedings* (Goslar, 2014), ... B. B., (Ed.), Eurographics Association, pp. 121–130. doi:10.2312/3dor.20141058. 1
- [LRS10] LIAN Z., ROSIN P. L., SUN X.: Rectilinearity of 3D Meshes. *International Journal of Computer Vision* 89 (2010), 130–151. doi:10.1007/s11263-009-0295-0. 4
- [MFP\*13] MACHADO J., FERREIRA A., PASCOAL P. B., ABDELRAHMAN M., AONO M., EL-MELEGY M., FARAG A., JOHAN H., LI B., LU Y., TATSUMA A.: Shrec'13 track: Retrieval of objects captured with low-cost depth-sensing cameras. In *Proceedings of the Sixth Eurographics Workshop on 3D Object Retrieval* (Aire-la-Ville, Switzerland, Switzerland, 2013), 3DOR '13, Eurographics Association, pp. 65–71. URL: <http://dx.doi.org/10.2312/3DOR/3DOR13/065-071>, doi:10.2312/3DOR/3DOR13/065-071. 1
- [MR08] MUTHUGANAPATHY R., RAMANI K.: Shape retrieval contest 2008: Cad models. In *Shape Modeling and Applications, 2008. SMI 2008. IEEE International Conference on* (June 2008), pp. –. doi:10.1109/SMI.2008.4547977. 1
- [OFCD02] OSADA R., FUNKHOUSER T., CHAZELLE B., DOBKIN D.: Shape Distributions. *ACM Transactions on Graphics* 21 (2002), 807–832. 2
- [PR99] PAQUET E., RIOUX M.: Nefertiti: a query by content software for three-dimensional models databases management. In *Proceedings of the International Conference on Recent Advances in 3-D Digital Imaging and Modeling* (Washington, DC, USA, 1999), NRC '99, IEEE Computer Society, pp. 345–. URL: <http://dl.acm.org/citation.cfm?id=523428.825366>. 2
- [SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The Princeton Shape Benchmark. *Shape Modeling and Applications, International Conference on 0* (2004), 167–178. doi:<http://doi.ieeecomputersociety.org/10.1109/SMI.2004.1314504>. 1, 2, 3
- [SSG\*03] SUNDAR H., SILVER D., GAGVANI N., DICKINSON S., Y D. S.: Skeleton Based Shape Matching and Retrieval, 2003. 2
- [TV08] TANGELDER J. W., VELTKAMP R. C.: A survey of content based 3D shape retrieval methods. *Multimedia Tools Appl.* 39, 3 (2008), 441–471. URL: <http://portal.acm.org/citation.cfm?id=1395016.1395041>, doi:10.1007/s11042-007-0181-0. 2