Eurographics Workshop on 3D Object Retrieval (2011)
H. Laga, T. Schreck, A. Ferreira, A. Godil, I. Pratikakis, and R. Veltkamp (Editors)

A GPU based high-efficient and accurate optimal pose alignment approach of 3D objects

Qian Zhang   Jinyuan Jia

School of Software Engineering, Tongji University
Cao'an Highway, No.4800, Shanghai, China, 201804

Abstract

In this paper we present a new method for alignment of 3D objects. This approach is based on the exhaustive optimization search in the 3D space using GPU based genetic algorithm. The descriptor of 3D object used as the objective function to be optimized is a newly developed pose-variant similarity measure, which is obtained directly from the voxelized model's geometry and could be entirely implemented on the GPU. In order to reduce the traditional optimal algorithms' large processing time, we exploit the GPU's highly parallel architecture and transport our approach from CPU to GPU. Experimental results show that the proposed method is superior to existing normalization techniques such as PCA and provides a high degree of precision to align 3D objects.

1. Introduction

Pose normalization of 3D models is a common preprocessing stage in many applications in computer graphics such as visualization, 3D object recognition, 3D model matching and retrieving [SMKF05, TV08]. 3D objects are generally given in arbitrary scale, position and orientation in 3D space, then it is very essential to normalize the models into a canonical coordinate frame before any further processing. The most well-known approach computing the alignment of 3D objects is the principal component analysis(PCA) method [PRMN00, VSR02] which based on the computation of moments of the 3D objects. After a translation of the center of mass to the origin of the coordinate system, three principal axes computed with PCA are used to determine the orientation. However, principal axes often feature severe inaccuracies regarding the alignment of similar objects, which is illustrated in [FMK*03, BKS*05]. In most cases, 3D models are represented as basic geometric meshes and attributions of the meshes are used as input for PCA, such as vertices and normals. Since these information may not be uniformly distributed over the surfaces which may reduce the quality of the PCA, various methods to solve this defect have been proposed. Paquet [PRMN00] et al. weights each vertex of 3D meshes by average area of the surrounding triangles. In [AKKP05], Abfalg et al. generate uniformly distributed sampling points on the object's surface before PCA is applied. Zhang [ZJL10] et al. also introduced the voxels based PCA which is insensitive to the quality of 3D meshes and improved the robustness of PCA's alignment results in some degree.

On the other hand, some optimization methods define a measure of similarity between 3D objects and search for the rotation which maximizes or minimizes this similarity function. An example for such approach is presented by Kazhdan [Kaz07], it decompose the 3D optimization alignment problem into tow sub-problems of 2D and 1D optimization aligning respectively. In [CVB08], Mohamed Chaouch et al. presented an symmetry properties based methods to get the optimal alignment of 3D objects, it used the good properties of PCA techniques with respect to the planar reflective symmetry in spatial space. In [MG09], Martinek et al. introduced a GPU based optimal 3D object alignment approach, it is similar with Kzahdan's method. They proposed a GPU based similarity function using improved depth images to accelerate approximately optimal alignment searching and implemented the approach on GPU platform to obtain high efficiency.

2. The overall scheme

Our approach is an optimization method too but differ with the previous ones. The overall scheme of our approach is described in Fig.1. Firstly, each 3D model is represented by a new similarity function. Then the genetic algorithm is used to determine the optimal pose that minimizes the distance between two models. Our approach transforms the 3D

DOI: 10.2312/3DOR/3DOR11/097-100

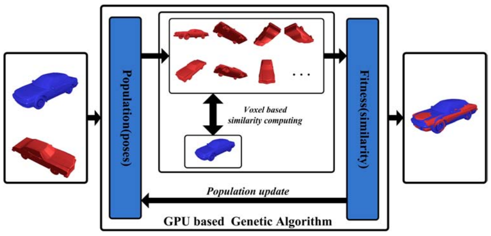


Figure 1: *The overall scheme of our proposed methodology.*

alignment problem into an optimal solution searching in 3D space. A GPU based genetic algorithm is designed for such an optimal pose searching, which considers rotation, translation and scale synthetically to improve the quality of 3D alignment.

3. Voxel-based similarity measuring

3.1. Pose representation

Basically, there are three transformations, rotation, translation and scale for posing a 3D model in 3D space. Translation is taken in terms of the center of a 3D model and described by a spatial vector: $\boldsymbol{t} = \{tx, ty, tz\}$. Scaling changes the size of a 3D object and represented by a numeric parameter: $\boldsymbol{s} = \{s\}$. Rotation is the most important operation for 3D alignment and could be represented by using a triplet of *Euler* angles $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and $\phi, \varphi \in [-\pi, \pi]$, which is similar with [Kaz07]. This triplet expresses every orientation of the pose as a rotation about the y-axis, multiplied on the left and right by rotation about the z-axis, given by $\boldsymbol{r}(\phi, \theta, \varphi)$. Now, a pose of the object B in 3D space can be formulated by P:

$$P = \{\boldsymbol{r}, \boldsymbol{t}, \boldsymbol{s}\}$$
$$\Rightarrow P(\boldsymbol{r}, \boldsymbol{t}, \boldsymbol{s}) = \begin{bmatrix} c_\phi c_\varphi + s_\phi s_\theta s_\varphi & s_\varphi c_\theta & -s_\phi c_\varphi + c_\phi s_\theta s_\varphi & 0 \\ -c_\phi s_\varphi + s_\phi s_\theta c_\varphi & c_\varphi c_\theta & s_\varphi s_\phi + c_\phi s_\theta c_\varphi & 0 \\ s_\phi c_\theta & -s_\theta & c_\phi c_\theta & 0 \\ tx & ty & tz & s \end{bmatrix} \tag{1}$$

where $c_\phi = \cos\phi, s_\phi = \sin\phi, c_\varphi = \cos\varphi, s_\varphi = \sin\varphi, c_\theta = \cos\theta, s_\theta = \sin\theta$. Notice the fact that both two 3D object are normalized at first, then the domain of the translating and scaling are restricted within a sub-domain of the whole 3D space. If a translation vector moves the object B out of the *AABB* of 3D object A, then, these two models will never overlap and are unnecessary to do alignment again. Thus, the translation vector can be limited within the domain $\{x, y, z | x, y, z \in [-1.0, 1.0]\}$. As to scaling, if the object B is too small or too large, then the alignment effect will become rather bad obviously, so the domain of scaling part of a pose can be confined within $[0.5, 2.0]$.

3.2. Voxel-based similarity function

In our pose alignment approach, a voxel-based similarity function for 3D object is proposed. Initially, the tightest *AABB* circumscribing the 3D model is constructed, then both the model and this *AABB* are normalized to norm-size to scale the invariance. Decompose this bounding box into $n \times n \times n$ uniform voxels grid and set each voxel with the initial state as transparent. Hence, we use $voxel(x, y, z, s)$ to describe the state of one voxel, where (x, y, z) is the spatial position of this voxel and s is its size. For a initial voxel located at (x, y, z) is regarded as an opaque voxel, notated as $voxel(x, y, z, 1) = 1$, if there are geometric primitives located within this voxel; otherwise, it is regarded as transparent, notated as $voxel(x, y, z, 1) = 0$. All the opaque voxels form a collection to represent geometric features of 3D object for high efficient and accurate similarity computing:

$$\boldsymbol{C} = \{voxel(x, y, z, s) | voxel(x, y, z, s) = 1\} \tag{2}$$

where $1 \le x, y, z \le n$. Denote the voxels collections of the objects A and B as A_C and B_C, respectively. Define a $voxel(x, y, z, s)$ belong to the intersection set $A_C \cap B_C$ only if this voxel(assuming it belong to A_C) intersect with at least one voxel in the voxels collection of B_C. Then a good measure of how much these two objects match each other from the initial pose is the ratio:

$$S = \frac{|A_C \cap B_C|}{|A_C \cup B_C|} \tag{3}$$

For each potential pose in 3D space of object B, the spatial position must be applied to object B to update the similarity function. Given an initial voxel $\boldsymbol{v}$ belonged to object B whose current pose is P, its initial status is $\{x, y, z, 1\}$, then the new status $\boldsymbol{v}'$ of $\boldsymbol{v}$ can be obtained by:

$$\begin{aligned} \boldsymbol{v}' &= \boldsymbol{v}P = [x, y, z, 1]P \\ \Rightarrow \boldsymbol{v}' &= [x', y', z', s] \end{aligned} \tag{4}$$

where (x', y', z') is the updated position of $\boldsymbol{v}$ under pose P, s is its updated size respectively. Through this operation, the pose of object B could be easily and independently applied, which become the basic of our GPU based similarity measuring and optimal pose searching. Since each voxel is a spatial axis-aligned cube with different size, the similarity computing become the intersection testing between these two voxels collections of the two objects. The whole voxels set of one object could also be divided into small parts to do the intersection testing independently, this make our approach high parallelized and convenient to implement on GPU.

4. Optimal alignment of 3D objects

The problem of finding the optimal alignment between two 3D objects now can be formulated as a global optimization problem, where the task is to determine the pose P_{max} of object B so that $\Omega(P_{max}) = max(\Omega(P))$ within the confined 3D space. Traditional blinded searching in the potential solution domain is rather slow and inefficient, even the domain of potential solution is a subset of the infinite 3D space.

One more efficient way to accelerate the searching is using heuristic optimal algorithms, here, we chose and designed an improved genetic algorithm to achieve this purpose.

4.1. GPU based genetic algorithm

Genetic algorithms(GAs) [Hol92] are powerful, domain-independent search techniques inspired by Darwinian theory. An typical GA always employ selection, mutation and crossover to generate new search points in a state space. Although GAs are effective in solving optimal problems, the large execution time are their fatal weakness. In this work, we designed and implemented a GPU based GA(GPUGA) with remarkable short running time to do the optimal 3D alignment searching. The brief architecture of our GPUGA is shown in Fig.2.

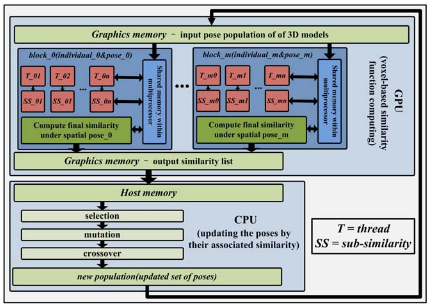


Figure 2: *The structure of our GPU based genetic algorithm. The reasonable and adequate parallelization are its most important features.*

4.2. Heuristic initial poses

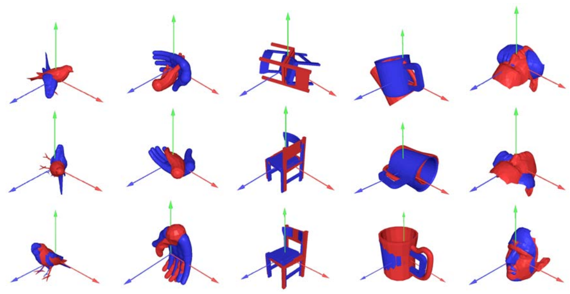
Furthermore, some heuristic strategies are also employed to produce better initial population of GPUGA. It will reduce the iteration of GPUGA's evaluation and make the searching terminate more sooner. For the pose P of object B, there are several potential initialization statuses which are more nearby the optimal poses in global space to obtain the optimal alignment:

1. The initial pose of object B,
$$P = \{\boldsymbol{r}(0,0,0), \boldsymbol{t}(0,0,0), \boldsymbol{s}(1)\} \tag{5}$$
The initial pose of a 3D object may be accordance with the perceptive system of human being, thus using these initial poses to alignment models may obtain good alignment result.
2. The pose of object B in terms of its PCA coordinate systems referring to the pose of A in its PCA systems respectively,
$$P = P^A_{pca} \cdot P^B_{pca} \tag{6}$$
PCA is a fast and efficient way to approximately align 3D model into canonical coordinate frame, which could acquire the good aligning pose and accelerate the search of optimal alignment process.
3. Some special rotating pose of object B, such as obtained by rotating along the x,y,z axes with angles $\{\frac{\pi}{2}, \pi, 2\pi, ...\}$,
$$P = \{\boldsymbol{r}(\alpha, \beta, \gamma), \boldsymbol{t}, \boldsymbol{s}\} \tag{7}$$
where $\alpha, \beta \in \{\frac{\pi}{2}, \pi, 2\pi\}, \gamma \in \{\frac{\pi}{2}, \pi\}$. 3D objects always be applied several spatial transformations by artists or programmers before used, but most of these operations are only axes's direction changing or axis-aligned rotating. Hence, the poses to inverse these operations are gathered to form the thirdly initial poses.

These initial poses are produced heuristically in certain probability and then inserted into the initial population of GPU based genetic algorithm.

5. Experimental results

To evaluate the efficacy of our approach, we restrict our comparison to two widely used PCA normalization techniques: CPCA [VS03] and NPCA [PPPT07]. First of all, an optimal alignment standard need to be generated to test the performance objectively. Here, we set the size of initial population and max generation of GA to be large enough to obtain the approximately global optimal alignment as the comparing standard. On the other hand, in order to test the robustness of our voxel-based similarity function, the depth-images based similarity function are used to compute similarity in standard generating. In our experiment, six axis-aligned depth images surrounding the 3D object were generated to compute the similarity between models [VKTP04]. The optimal pose standards of alignment used in our experiment are obtained under following parameters: the size of depth images are 256×256; the max generation of the GA is 1000; the population's size of GA is 1024.



(a) Bird (b) Hand (c) Chair (d) Cup (e) Face

Figure 3: *Some alignment examples of our approach comparing with the PCA methods. From top to down aligned by CPCA, NPCA and our method respectively.*

The comparison results of our experiment in Table 1 show that our approach achieve better alignment effects obviously than both CPCA and NPCA. Also, our proposed voxel-based similarity function has a strong discrimination ability of 3D objects. Some alignment examples can be found in Fig.3, it explains the unsatisfactory worst-case results of PCA and shows our proposed approach could provides a better alignment.

Class	CPCA(%) Average/worst case	NPCA(%) Average/worst case	Our method(%) Average/worst case
Airplanes	81.7/32.4	82.4/30.1	96.4/93.8
Horses	55.4/31.6	56.9/29.7	93.1/88.4
Cars	91.4/74.0	86.7/67.3	99.4/92.5
Cups	75.7/37.1	81.6/67.4	89.2/78.5
Humans	91.5/72.4	85.8/70.1	98.9/94.6
Tables	76.2/23.4	71.8/31.7	94.3/91.5
Chairs	67.9/29.5	71.7/30.2	96.5/84.3
Hands	51.5/32.7	54.5/33.9	81.4/71.1

Table 1: *The alignment results tested on several typical classes of 3D models from PSB. The resolution of voxelization is* $128 \times 128 \times 128$*, the size of population of GA is 256 and the max generation is 150.*

The performance data of our approach are listed in Table 2, which show that the alignment quality of two 3D models depends on the resolution of voxelization, especially, when the resolution becomes higher than 128, the alignment results perform promising, however, its running time is rather short and acceptable. Therefore, our proposed method can be employed in various applications such as online 3D object retrieving and matching.

Voxel-resolution	CPU(sec) Average	GPU(sec) Average	Speedup Average	Accuracy(%) Average/worst
32	71.179	0.470	151.582	33.7/17.3
64	289.718	1.051	275.705	69.5/51.6
128	679.578	1.785	380.686	89.5/69.0
256	2652.107	4.741	432.824	94.4/77.9

Table 2: *The performance data of our approach tested on different platforms and parameters.*

6. Conclusion and future work

In this paper, we proposed an efficient and accurate approach for alignment of 3D objects. It is based on the exhaustive optimization search in 3D space to find the optimal aligning pose between two 3D models, which GPU based genetic algorithm is implemented to searching and a new voxel-based similarity function were proposed to do the similarity computing and fitness updating. Some heuristic strategies were also used to narrow the searching domain and speedup the optimal pose searching. The experimental results tested on Princeton Shape Benchmark shown that our proposed method is superior to existing PCA techniques in terms of precision. In the near future, we will further investigate other shape descriptor of 3D objects to improve robustness, precision and efficiency, also, more experiment against other approach will be implemented.

Acknowledgement

This work was partially supported by Shanghai grant of key breakthrough of science and technology(No.08511501001).

References

[AKKP05] ABFALG J., KRIEGEL H., KROGER P., POTKE M.: Accurate and efficient similarity search on 3d objects using point sampling, redundancy, and proportionality. *Advances in Spatial and Temporal Databases* (2005), 200–217. 1

[BKS*05] BUSTOS B., KEIM D., SAUPE D., SCHRECK T., VRANIĆ D.: Feature-based similarity search in 3D object databases. *ACM Computing Surveys(CSUR) 37*, 4 (2005), 345–387. 1

[CVB08] CHAOUCH M., VERROUST-BLONDET A.: A novel method for alignment of 3D models. In *Proceeding of IEEE International Conference on Shape Modeling and Applications* (2008), IEEE, pp. 187–195. 1

[FMK*03] FUNKHOUSER T., MIN P., KAZHDAN M., CHEN J., HALDERMAN A., DOBKIN D., JACOBS D.: A search engine for 3D models. *ACM Transactions on Graphics 22* (2003), 83–105. 1

[Hol92] HOLLAND J.: *Adaptation in natural and artificial systems*. MIT Press Cambridge, MA, USA, 1992. 3

[Kaz07] KAZHDAN M.: An approximate and efficient method for optimal rotation alignment of 3D models. *IEEE transactions on pattern analysis and machine intelligence* (2007), 1221–1229. 1, 2

[MG09] MARTINEK M., GROSSO R.: Optimal rotation alignment of 3D object using a GPU-based similarity function. *Computer and Graphics 33*, 3 (2009), 291–298. 1

[PPPT07] PAPADAKIS P., PRATIKAKIS I., PERANTONIS S., THEOHARIS T.: Efficient 3D shape matching and retrieval using a concrete radialized spherical projection representation. *Pattern Recognition 40* (2007), 2437–2452. 3

[PRMN00] PAQUET E., RIOUX M., MURCHING A., NAVEEN T.: Description of shape information for 2D and 3D objects. *Signal Processing: Image Communication 16*, 1-2 (2000), 103–122. 1

[SMKF05] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The Princeton Shape Benchmark. In *Proceeding of Shape Modeling Applications* (2005), IEEE, pp. 167–178. 1

[TV08] TANGELDER J., VELTKAMP R.: A survey of content based 3D shape retrieval methods. *Multimedia Tools and Applications 39* (2008), 441–471. 1

[VKTP04] VAJRAMUSHTI N., KAKADIARIS I., THEOHARIS T., PAPAIOANNOU G.: Efficient 3D object retrieval using depth images. In *Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval* (2004), ACM, pp. 189–196. 3

[VS03] VRANIC D., SAUPE D.: 3D model retrieval. In *Proceedings of Spring Conference on Computer Graphics and its Applications* (2003), Citeseer, pp. 3–6. 3

[VSR02] VRANIC D., SAUPE D., RICHTER J.: Tools for 3D-object retrieval: Karhunen-Loeve transform and spherical harmonics. In *Proceeding of IEEE Fourth Workshop on Multimedia Signal Processing* (2002), IEEE, pp. 293–298. 1

[ZJL10] ZHANG Q., JIA J., LI H.: A GPU based 3D Object retrieval approach using spatial shape information. In *Proceedings of the IEEE International Symposium on Multimedia* (2010), pp. 212–219. 1