Retrieving Matching CAD Models by Using Partial 3D Point Clouds

Cheuk Yiu Ip[1] and Satyandra K. Gupta[2]

[1]University of Maryland, ipcy@umd.edu
[2]University of Maryland, skgupta@umd.edu

ABSTRACT

The ability to search for a CAD model that represents a specific physical part is a useful capability that can be used in many different applications. This paper presents an approach to use partial 3D point cloud of an artifact for retrieving the CAD model of the artifact. We assume that the information about the physical parts will be captured by a single 3D scan that produces dense point clouds. CAD models in our approach are represented as polygonal meshes. Our approach involves segmenting the point cloud and CAD mesh models into surface patches. The next step is to identify corresponding surface patches in point clouds and CAD models that could potentially match. Finally, we compute transformations to align the point cloud to the CAD model and compute distance between them. We also present experimental results to show that our approach can be used to retrieve CAD models of mechanical parts.

Keywords: Shape Matching, 3D Scanning, CAD Database

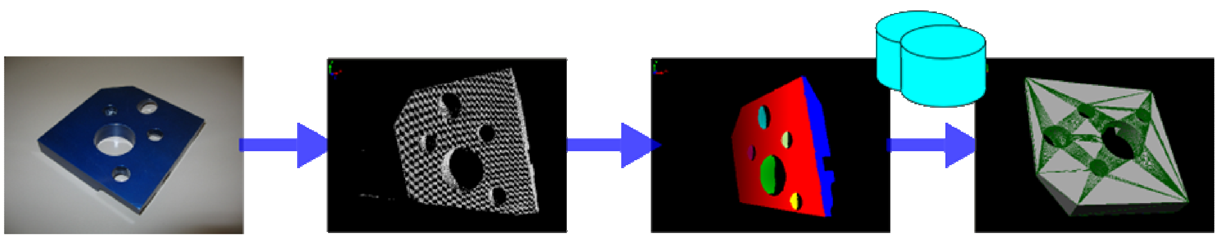


Fig. 1: An overview of the scan-to-CAD-search system.

1. INTRODUCTION

The ability to search for a CAD model that represents a specific physical part is a useful capability that can be used in many different applications. The following scenario illustrates the usefulness of being able to search for a CAD model based on point cloud generated by a partial scan. Let us assume that a part needs to be replaced in a complex machine. There is no label on the part. Hence the user does not know the part number. The user scans the physical part using a 3D scanner and generates the point cloud. This point cloud is then used by the user to search the CAD database and find the CAD model of this part. The CAD model has the information about the part number and the user is able to order the replacement part using the part number.

In order to initiate the search, one needs to describe of the desired physical part. 3D scanning can provide the models for initiating the search. A single part scan only takes a few seconds. However, to scan a part completely by using optical digitizing instruments can be a time consuming process, because it often requires a large number of scans to complete the acquisition from multiple sides. Each scan can only cover one side, practically about 150 degrees, of the target part. Furthermore, occlusions create holes on the resulting point cloud. Hence, it may be necessary to scan the same side from multiple angles to resolve any uncovered area. Lengthy post-processing is also required to remove noise, register, merge, and triangulate the point clouds to form a complete model. Hence, we believe that building a complete part scan is not practical in this application due to registration difficulties and increase in the scanning setup complexity. We assume that the information about the physical parts will be captured by a single 3D scan (also called partial scan because it only captures a portion of the part's boundary) that produces dense point clouds. Hence, we are interested in developing an approach that can work with partial part scans.

In order to meet the application requirements, the search algorithms have to have the following characteristics.

- There might be many similar parts in the database that may differ only very slightly in terms of feature and dimensions. Hence the algorithm has to be precise enough to find the right CAD model as a match and successfully reject the CAD model of the very similar parts.

- The algorithm has to be computationally fast enough to search through a large database. Hence, approaches based on global registration of point clouds to CAD models are not likely to work well in this application.

- Finally, the scan may produce partial point clouds for some faces. Hence the algorithm cannot make any strong assumptions about the completeness of the point cloud.

Previous research on CAD model retrieval was focused on locating similar models by using their gross shape information. These approaches often first compute shape descriptors of parts by extracting representative features from the gross shape of the models, and then subsequently compare the descriptors to evaluate the similarities. To accurately compute the shape descriptors, it often requires complete models of the query and database objects. Hence these methods do not work well for meeting the above three requirements.

This paper presents an approach to use partially scanned 3D point cloud of an artifact for retrieving the CAD model of the artifact. In this paper, we introduce an approach based on partial matching to support retrieval of CAD models based on partial point clouds. CAD models will be represented as polygonal meshes. Hence, we will use term CAD mesh model to refer to faceted CAD models. Our system is designed to match point clouds, acquired by a single 3D scan, to complete CAD mesh models. This is accomplished through a segmentation procedure and local matching. Our approach consists of the following steps: (1) segmenting the point cloud and CAD model into similar sets of surfaces patches using the same algorithm, then (2) matching up corresponding patches according to their properties, and (3) computing the possible transformations and evaluate the matching error. Fig. 1 shows schematically how the proposed approach will work.

Based on the approach outlined above, we have built a scanning-based-shape-search system that compares partial point clouds for mechanical parts to their CAD models and enables users to retrieve a CAD model that matches a given point cloud. The rest of this paper is organized in the following manner. Section 2 presents a brief review of related work. Section 3 describes our problem formulation. Section 4 explains the details of our approach. Section 5 demonstrates how our system works by retrieving CAD models using synthetic and scanned point clouds. Section 6 presents the concluding remarks and discusses possible future work.

2. RELATED WORK

2.1 Comparing Shape Models of CAD

In this paper, we focus on the matching of point clouds to CAD models. Most CAD models are solid models that are defined parametrically. Due to the development of rapid prototyping and visualization areas, approximate shape models represented by a polygonal mesh and dense point clouds are becoming another useful alternative to CAD representations. As mentioned earlier, we will use polygonal mesh models as the approximation of CAD models in our work.

Shape model representations of 3D objects are approximate models characterized by a mesh of polygons or a cloud of points for presentation or rendering purposes in computer graphics. Rather than exact parametric equations, polygons or densely sampled points are used to approximate curved surfaces. Only the geometry of triangles and points are stored without any topological information. In contrast to proprietary solid model formats, open mesh file formats such as VRML, STL, and ASCII point clouds are widely available. Although shape models are not suitable for many tasks in CAD/CAM systems, polygonal meshes can serve as the lowest common denominator in comparing CAD models. CAD mesh models can be generated by faceting solid models from different modeling systems. Shape models of objects can also be acquired easily by using 3D scanners or CT to enable comparison of digital and physical artifacts.

From the polygon mesh, different transformation invariant attributes can be extracted as the means of similarity among 3D models. Thompson et al. [28] examined the reverse engineering of designs by generating surface and machining feature information off of range data collected from machined parts. The method of Osada et al. [24] creates an abstraction of the 3D model as a probability distribution of samples from a shape function acting on the model. Novotni and Klein [23] demonstrated the use of 3D Zernike descriptors. Kazhdan et al. [20] compared 3D models with spherical harmonics.

While these techniques target general 3D models, Ip et al. [14, 15] focused on comparing shape models of CAD with shape distributions. Iyer et al. [17] presented a CAD oriented search system, based on shape, voxelization and other approaches. Pal et al. [25] extracted features from CAD models using genetic algorithms. Cardone et al. [4] compared prismatic machined parts by using machining features. Various database techniques for CAD are discussed in [6, 7, 12].

Recently, research efforts in industry and academia are examining the use machine learning techniques to train a 3D shape recognition system with CAD data. Work in industry has explored the use of neural networks to identify parts based on multiple 2D views [27]. Hou et al. [13] attempted to use shape information to cluster the semantics of parts with SVMs. In the context of shape model matching, Elad [8] used linear SVMs to adjust retrieval results from a 3D shape database according to users' feedback. Ip et al. [16] classified models according to manufacturing processes by a curvature descriptor and SVMs.

There are recent approaches that employ partial matching of models. Bespalov et al. [3] used scale-space representations to segment different features of meshes. Funkhouser et al. [9] partially matched shape features according to different priorities. More extensive surveys and literature reviews in this area can be found in references [5], [18], and [29].

2.2 Point Cloud Alignment and Registration

The availability of 3D scanning technologies (Laser, white light, and CT scanners) has stimulated the interest in 3D point cloud alignment and registration. Given two point clouds with overlapping regions, registration based on iterative closest points (ICP) aims to rotate and translate a point cloud to match the other one. Because laser scanners and range finders often come with limited measure volume, registration becomes a critical process when acquiring 3D images of large scale parts in the industry. Since Besl et. al [2] published the original ICP algorithm, there have been many variations with different kind performance improvements in some of the recent work. Rusinkiewicz et al. [26] published a survey of the ICP techniques and demonstrated a fast variant that registers point clouds in real time. Mitra et al. [22] optimized the registration according to the point cloud geometry. Gefland et al. [11] proposed a method to find a good initial alignment of overlapping point clouds in an arbitrary orientation for ICP.

2.3 Mesh Segmentation

Research in partitioning triangular meshes into separated meaningful surface patches is of great interest for many applications, such as, shape simplification, compression, analysis, and recognition. We briefly review some of the more recent approaches. Attene et al. [1] recently published a comparative study on recent mesh segmentation techniques. Mangan et al. [21] applied computer vision style watershed method to segment surfaces according to total curvature. Yamauch et al. [30] segmented surfaces with mean shift algorithm. Hierarchical decomposition is another popular approach. Garland et al. [10] introduced hierarchical face clustering. Katz et al. [19] used fuzzy clustering and cut to decompose triangular meshes.

3. PROBLEM FORMULATION

This paper describes an approach to locate a CAD model in a database by using a partial scan of the underling artifact. In the subsequent description, a part is denoted by P , its point cloud is P_s , and the corresponding CAD model is P_m . Acquiring P_s by scanning P is very similar to evenly sample points on the surface of P_m . A partial scan of P , is denoted by P_{ps}, where $P_{ps} \subseteq P_s$. The goal is to align P_{ps} with respect to P_m, such that, the distance of P_{ps} and P_m can be minimized. The distance between aligned P_{ps} and P_m can be used to determine if P_{ps} matches P_m. Since $P_{ps} \subseteq P_s$, all of P_{ps} must be lying on some parts of P_m. Since $P_{ps} \cap P_s = P_{ps}$, it is not necessary to identify the overlapping points. This subset assumption eliminates one of the hardest problems in general point cloud registration. The matching quality in between the point cloud and the part is evaluated by the statistics of distances in between every point in P_{ps} and the surface of P_m. As P_{ps} is just a partial scan, any uncovered area is assumed to be insufficient data rather than error.

4. TECHNICAL APPROACH

Partially scanned point clouds and polygonal CAD models (CAD meshes) are first separated into surface patches, then aligned and compared according to the principal components of the surface patches. Our approach of matching scanned point cloud to CAD meshes consists of three stages:

1. Segmentation of point cloud and CAD meshes into surface patches using an identical algorithm.
2. Identification of the matching patches in point cloud and CAD meshes.
3. Aligning the point cloud with the CAD meshes and evaluating the error associated with the alignment.

In the approach presented in this paper, point clouds are assumed to be evenly sampled on the target surface. This assumption is consistent with the raw data produced by many popular 3D scanners.

4.1 Segmentation of Point Cloud and CAD Mesh into Surface Patches

Point clouds and CAD meshes are segmented into surface patches using an identical algorithm. It is important to apply the same approach to both the point cloud and the CAD mesh to ensure the similar surface patches are produced from the matching point cloud and CAD mesh. Our approach segments the point clouds and CAD meshes according to curvature for comparisons. Partial matching of 3D models is a challenging problem for many global shape descriptors. The shape of a partial scan often differs from its complete scan counterpart, e.g. the change of total length, width, and height. Hence, many global shape descriptors will discriminate a model against its own fragments. In addition, many 3D scans are imperfect. Hence, lengthy post-processing is often required to fill holes and remove noises from the point cloud. In attempt to alleviate these issues, we first segment the point clouds and CAD meshes into local patches and use them as matching units. This approach removes the gross shape dependency problem by separating both the partial scan and the CAD meshes into similar local surface patches that can directly be compared. Any extra patches from the CAD mesh will be ignored during evaluation. The segmentation procedure also allows us to discard insignificant patches, which are possibly noise, from the scanned point cloud.

The surface patches of point clouds and meshes are created according to their surface curvature values. This simple method is generally sufficient to partition CAD surfaces. For complex freeform surfaces, more sophisticated or semantic based segmentation algorithm may be required in future. Curvature defines the variation of surfaces patches and it is a popular criterion among many previous segmentation approaches. The identical segmentation algorithm is applied to both point clouds and CAD meshes. This allows similar patches to be generated on corresponding point clouds and CAD meshes. It is very important to ensure the patches of the matching point clouds and meshes are close enough. These patches will be used as matching primitives and they will be compared with one another. Since the surface patches are similar, it is not necessary to perform many-to-many matching on the surface patches.

Total curvature is computed from the normal vectors distribution of local neighborhoods on the surface. Normal vectors on the mesh model are sampled according to the mesh connectivity, for smooth meshes, normal vectors in a 1-ring neighborhood are sufficient for curvature computation. At the same time, normal vectors on the point cloud are estimated by normals of the best fitted planes of small neighborhoods of points. Following the method described in [21], the total curvature of a small neighborhood can be estimated by the norm of the covariance matrices of its normal vectors. Neighboring points and triangles that share similar curvature are grouped into patches. Fig. 2 shows a side by side segmentation comparison of the patches identified in the point cloud and CAD mesh for Part A.

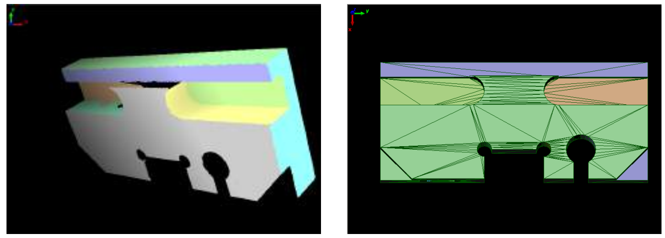


Fig. 2: Corresponding segmented point cloud and CAD mesh model for Part A.

4.2 Matching Point Cloud and CAD Patches

The correspondence of matching point cloud and CAD patches are determined by some rotationally independent properties. This process aims to eliminate irrelevant matching patch pair candidates, especially, when none of the patches are similar. Given surface patches that are generated by the same segmentation algorithm, and P_{ps} is completely covered by P_m, if the point cloud and the mesh do not share any matching patches, the procedure can safely reject the CAD mesh and terminate.

Simple rotationally independent attributes such as surface area and curvature are used in our implementation. Only patches with both matching surface area and curvature will be considered for alignment. Patches in point cloud and CAD mesh are sorted first by their surface area. Then for each point cloud patch, the matching CAD mesh patch can be found by binary searching for mesh patches with the similar surface area. In our experience, large patches are more stable than smaller ones, as they are more likely to influence the shape. The resulting matching patches lists are then again sorted in descending order according to the surface area.

These rotationally independent attributes are the key to determine the correspondence of point cloud and CAD patches. When the point cloud and CAD mesh shares no patch, the CAD mesh can be eliminated at this stage, hence improve the overall performance by faster rejections. We only consider the largest k point cloud patches for alignment. Fig. 3 shows an example of point cloud and CAD mesh patches pairs, in this example $k = 4$. Hence, only four largest patches from the point cloud are considered. Smaller patches are avoided as they often represent noises and surfaces of standard features, such as holes and slots. The larger patches generally connect these features, hence representing discriminating patterns for different parts.

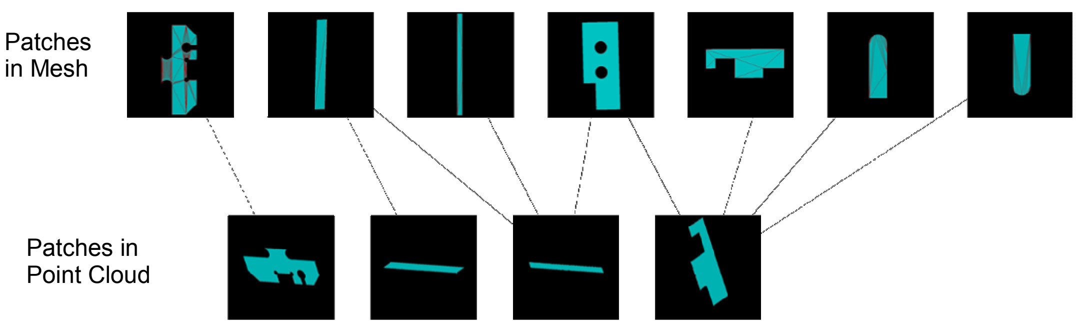


Fig. 3: Matching up point cloud and mesh surface patches.

4.3 Alignment of Point Cloud and CAD Mesh

Principal components of potentially matching patch pairs are computed to estimate possible transformation from the point cloud to CAD mesh. Principal components are dominating directions of the surface patches. When the pair of matching patches completely represents the same surface patch, they share a corresponding center of mass and the transformation between the patches will transform the point cloud to the CAD mesh.

The principal components are computed by analyzing the eigenvalues of the covariance matrices (cov) of the point cloud and CAD patches. Covariance matrices of a point cloud patch can be obtained by aggregating the distances in between the points to its center of mass, c_{ps}.

$$c_{ps} = \frac{1}{n}\sum_{n} p_{ps}, p_{ps} \in P_{ps}$$

$$\mathrm{cov}_{ps} = \frac{1}{n}\sum_{n}(p_{ps} - c_{ps})(p_{ps} - c_{ps})^{t}$$

Covariance matrices of a CAD mesh patch can be obtained by aggregating the distances in between the center of triangles to its center of area, c_m. The set of triangles of P_m is denoted by T_m, centers of P_m is denoted by p_m.

$$c_m = \frac{1}{area(T_m)}\sum area(t_m), t_m \in T_m$$

$$\mathrm{cov}_m = \frac{1}{n}\sum_{n}(p_m - c_m)(p_m - c_m)^{t}$$

The eigenvectors of the resulting decomposition are the principal components. To ensure that three component vectors form a right-hand coordinate system, the second principal direction is computed as the cross products of eigenvectors that associates with the largest and smallest eigenvalues. The eigenvectors that are associated with the smallest eigenvalue should be aligned to the normal direction of the patch. The two other component vectors may be flipped around the normal direction for 180 degrees. Two configurations of rotation matrices R_{ps} of the point cloud and R_m of the model can be composed by their respective sets of principle components. Both configurations should be tested when searching for the best alignment.

The rotation matrix:

$$R = R_m(R_{ps})^{t}$$

Translation vector:

$$T = c_m - Rc_{ps}$$

R and T transform the point cloud to the CAD mesh when they match. The distance in between the points in the point cloud and the CAD mesh is evaluated to measure the goodness of the alignment. When the point cloud matches the CAD mesh, the distance in between the transformed point cloud and the CAD mesh should be sufficiently small and the matching procedure returns true and terminates. While a large amount of point-to-CAD distance evaluation is computationally intensive, random sampling from the point cloud often provides a reasonable estimate of the average point-to-CAD distance.

This matching procedure generally terminates after evaluating the first few pairs of surface patches. If the point cloud and CAD mesh matches, the alignment of any correctly matched patches will approximately resemble the point cloud to CAD mesh transformation. The worst case scenario would be two mismatching parts that shares similar surfaces patches that are paired up for evaluation. To reduce matching time being spent on mismatching cases, as mentioned in the last section, only the largest k patches are considered for alignment. As the likelihood of proper alignment decreases along with the surface area of the surface patches pairs, the matching procedure can safely declare a mismatch of the point cloud and CAD mesh if the k largest surface patches do not match.

5. RESULTS

Experiments were conducted to assess the effectiveness of our approach in retrieving matching CAD models. We used both synthetic and real point clouds in our experiments.

For generating a synthetic point cloud, CAD models were randomly selected to be the query target. These models were manually rotated and translated to show representative features, then they were sampled for creating the query point cloud. Points are only sampled on certain triangles of the CAD model to mimic a real partial scan on the shell of a part, only triangles that are visible from one viewing direction (-z, viewing direction was used in the experiments) were considered. In this way, the synthetic data will realistically resemble a single scan of the part from the viewing direction, the resulting point cloud will comprise with the front face of the part as well as holes created by occluded regions (see Fig. 4). Dense points are repeatedly, randomly, and evenly sampled on triangles, until the average distance of points reaches 0.2mm, on average 200k points were sampled per point cloud. Gaussian noise (μ=0mm, σ=0.1mm) is added to the points along the viewing (z) direction. All datasets consisted of CAD models represented by triangular meshes. Only matching pairs that included the four largest point cloud patches were evaluated, the sample size of points was 10% of the dense point cloud. The matching criteria were (1) the average point to CAD distance was less than 1mm, and (2) the maximum point to CAD distance was less than five times of the average point to CAD distance. The second criterion tested if there are outliers during matching. This outlier test was necessary to reject very similar parts with minor variations.

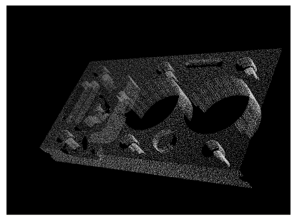


Fig. 4: Synthetic point cloud for Part B. The holes are occluded regions.

On the average, segmentation of query point cloud with 200k points takes 20 seconds. Segmentation of each CAD mesh, which can be computed offline, in the database took about 0.5 seconds. Matching surface patches took 0.0003 seconds computing point cloud to mesh transformation took 0.022 seconds. All the experiments were performed on a Linux platform running on a Celeron 1.6 GHz laptop equipped with 512MB of memory.

An experiment was performed to retrieve the CAD model that matches the point cloud from a group of heterogeneous models. This experiment aimed to test if the proposed approach would only retrieve the matching model, with no false positives, among dissimilar shapes. This dataset was provided by the National Design Repository at Drexel University [15]. It consisted of 55 prismatic machined parts of various shapes. The selected query point cloud, its corresponding CAD mesh, and point cloud to CAD alignment are shown in Fig. 5. Our approach correctly retrieved the exact matching CAD mesh. The average point cloud to CAD alignment distance was 0.39mm per point on the point cloud. The red segmented surface highlighted on the point cloud and CAD mesh denotes the matching surface patch. The exact match was returned as the only match by the system.

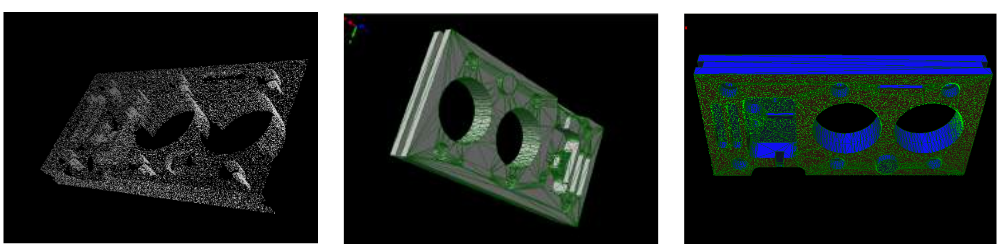


Fig. 5: Aligned point cloud with matching CAD model for Part B.

Another experiment was performed to retrieve the CAD models that match the point clouds from three different groups of similar models. This experiment aimed to test if only the exact match would be retrieved by our proposed approach. These three datasets provided parts that are very similar in gross shape but differ in minor details. For example, two parts shown in Fig. 6, are distinguished by the two holes and the two slots on their sides, otherwise they are identical. The three selected query point clouds, their corresponding CAD meshes, and point cloud to CAD alignments are shown in Fig. 7 (a), (b), and (c). Our approach correctly retrieved the exact matching CAD mesh. The average point cloud to CAD alignment distance was 0.7mm per point on the point cloud. The red segmented surfaces highlighted on the point cloud and CAD mesh in Fig. 7 (a), (b), and (c) denote the matching surface patches. The exact matches were returned as the only matches by the system.

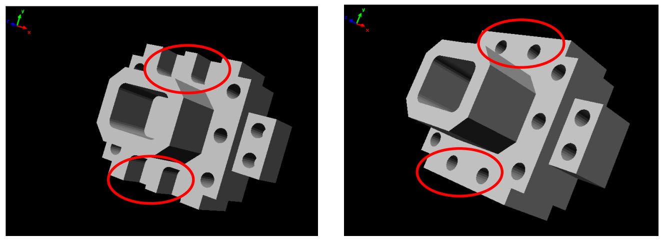
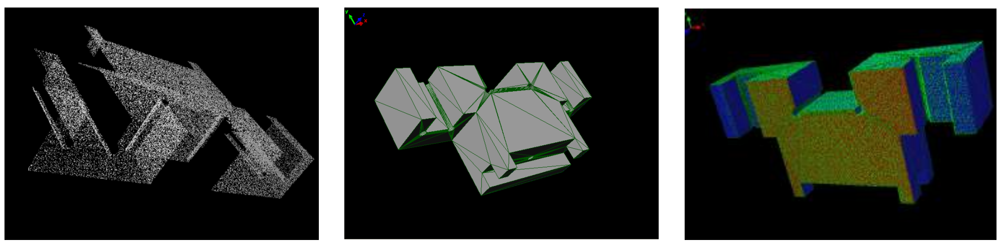
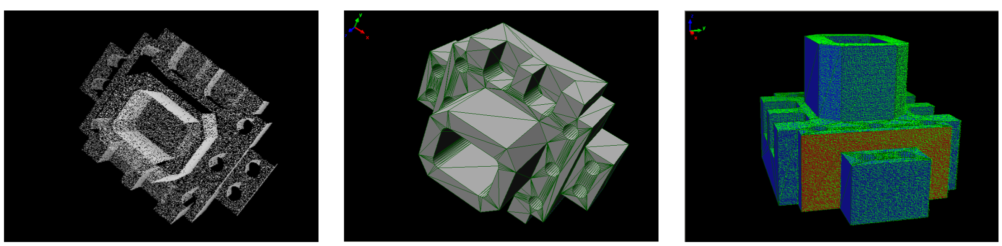


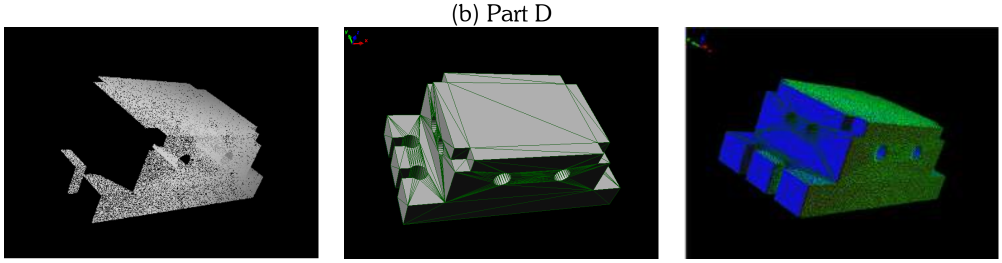
Fig. 6: Similar parts P586 and P755, red circles highlight their only differences.



(a) Part C



(b) Part D



(c) Part E

Fig. 7: Aligned point clouds and CAD models of test parts.

The last experiment used a point cloud of a physically scanned part to query the dataset presented in the first experiment and its matching CAD model. The test part is a standard CMM test part that comes with holes and slots in various sizes (see Fig. 8). The back face of it was scanned by a white light scanner for this retrieval experiment. Due to the occlusion of the holding fixture, the largest face could not be completely captured. The side surface turned out to be the only large enough patch, the red patch in Fig. 9, for matching. The segmented point cloud in Fig. 9 also shows the noise and outlier points were removed from the original point cloud (Fig. 8). The average point cloud to CAD alignment distance was 0.2mm per point on the point cloud. These results show our approach successfully found the matching patch, aligned the models, and retrieved the matching CAD model by using a point cloud.

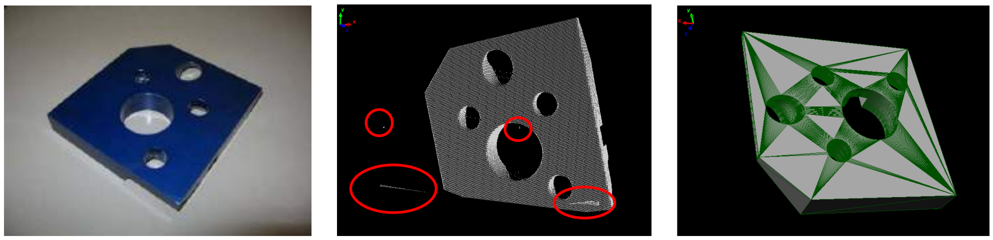


Fig. 8: A standard CMM test part, its point cloud, and its CAD model.

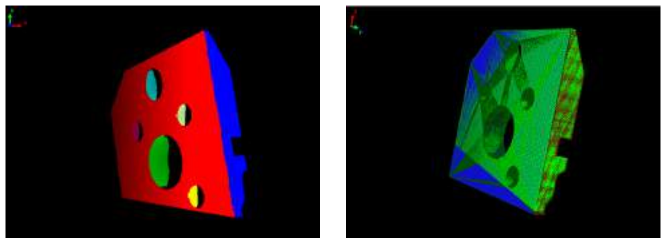


Fig. 9: The segmented point cloud, CAD aligned point cloud of the CMM test part.

6. CONCLUSIONS AND FUTURE WORK

This paper described a new approach to retrieve CAD models using partially scanned point clouds. The contribution of this research is the introduction of using a local surface based alignment to match incomplete dense point clouds to 3D mesh-based representations. This approach brings together 3D scanning and shape based CAD models matching and retrieval ideas. Point clouds acquired by 3D scanners can immediately be used as targets in CAD database queries. General shape matching challenges like rotational variance and incomplete shape information are resolved by the segmentation and local surface patches alignment processes. The experimental results have shown the proposed approach can locate exact matching models in various datasets. This shows that it is plausible to efficiently look up matching CAD models using 3D scanning.

To further accelerate the matching process, more rotational invariant attributes may be included during the patch matching stage. One alternative approach, introduced by [9], is to include a shape descriptor for each surface patch, while this is suitable for complex surface patches, it may be too complicated for engineering artifacts with only specific classes of surfaces. By including more discriminating attributes for specific surfaces, we believe the system's performance can further be tuned. On the other hand, as oppose to a fully automatic system, as presented in this paper, one may allow the users to interactively rank the importance of surface patches generated from point cloud. This changes the alignment order and may lead the system to discover an appropriate alignment faster.

7. ACKNOWLEDGEMENTS

This research has been supported in part by NSF grant DMI0093142. Opinions expressed in this paper are those of authors and do not necessarily reflect opinion of the sponsors.

8. REFERENCES

[1] Attene, M.; Katz, S.; Mortara, M; Patane, G; Spagnuolo, M; Tal, A: Mesh Segmentation - A Comparative Study, Shape Modeling International, 2006.

[2] Besl, P.; McKay, N.: A Method for Registering 3D Shapes, IEEE Transactions on Pattern Analysis and Machine Intelligence 14, 1992, 239-256.

[3] Bespalov, D.; Shokoufandeh, A.; Regli, W. C.; Sun, W: Scale-space representation and classification of 3d models, ASME Transactions, Journal of Computing and Information Science in Engineering, 3, 2003, 315-324.

[4] Cardone, A.; Gupta, S. K.; Deshmukh, A.; Karnik, K.: Machining feature-based similarity assessment algorithms for prismatic machined parts, Computer Aided Design, 38(9), 2006, 954-972.
[5] Cardone, A.; Gupta, S. K.; Karnik, K.; A survey of shape similarity assessment algorithms for product design and manufacturing applications, Journal of Computing and Information Science in Engineering, 3, 2003, 109--118.
[6] Chakraborthy, T.; Shape-based clustering of enterprise CAD databases, Computer Aided Design and Application 2(1-4), 2005, 145-154.
[7] Chu, C.-H.; Hsu Y.-C.: An integrated approach for 3D part search with multiple shape signatures, Computer Aided Design and Application 2(1-4), 2005, 183-192.
[8] Elad, M.; Tal, A.; Ar, S.**:** Content based retrieval of VRML objects: an iterative and interactive approach, Eurographics workshop on Multimedia, 2001.
[9] Funkhouser, T.; Shilane, P.: Partial Matching of 3D Shapes with Priority-Driven Search, Symposium on Geometry Processing, 2006.
[10] Garland, M.; Willmott, A.; Heckbert, P.: Hierarchical Face Clustering on Polygonal Surfaces, Symposium on Interactive 3D Graphics, 2001.
[11] Gelfand, N.; Mitra, N. J.; Guibas L.; Pottmann, H.: Robust Global Registration, Symposium on Geometry Processing, 2005.
[12] Gupta S. K.; Cardone, A.; Deshmukh, A.; Content-Based Search Techniques for Searching CAD Databases, Computer Aided Design and Applications, 3(6), 2006, 811-819.
[13] Hou, S.; Lou, K.; Ramani, K.: Svm-based semantic clustering and retrieval of a 3d model database, Computer Aided Design and Application 2(1-4), 2005, 155-164.
[14] Ip, C. Y.; Lapadat, D.; Sieger, L.; Regli, W. C.: Using shape distribution to compare solid models, Symposium on Solid Modeling and Applications, 2002.
[15] Ip, C. Y.; Sieger, L.; Regli, W. C.; Shokoufandeh, A.: Automated learning of model classifications, Symposium on Solid Modeling and Applications, 2003.
[16] Ip, C. Y.; Regli, W. C.: A 3D object classifier for discriminating manufacturing processes, Computer & Graphics, 30(6), 2006, 903-916.
[17] Iyer, N.; Jayanti, S.; Lou, K.; Kalyanaraman, Y.; Ramani, K.: Shape-based searching for product lifecycle applications, Computer Aided Design, 37(13), 2005, 1435--1446.
[18] Iyer, N.; Jayanti, S.; Lou, K.; Kalyanaraman, Y.; Ramani, K.: Three dimensional shape searching: State-of-the-art review and future trends, Computer Aided Design, 37(5), 2005, 509-530.
[19] Katz, S.; Tal, A.: Hierarchical Mesh Decomposition Using Fuzzy Clustering and Cuts, ACM Transactions on Graphics, 22(3), 2003, 954-961.
[20] Kazhdan, M,; Funkhouser,T,; Rusinkiewicz, S,: Rotation invariant spherical harmonic representation of 3d shape descriptors, Symposium on Geometry Processing, 2003, 156-164.
[21] Mangan, A. P.; Whitaker, R. T.: Partitioning 3D Surface Meshes Using Watershed Segmentation, IEEE Transactions on Visualization and Computer Graphics, 5(4), 1999, 308-321.
[22] Mitra, N. J.; Gelfand, N.; Pottmann, H.; Guibas, L.: Registration of point cloud data from a geometric optimization perspective, Symposium on Geometry Processing, 2004.
[23] Novotni, M.; Klein, R.: Shape retrieval using 3d Zernike descriptors, Computer Aided Design, 36(11), 2004, 1047-1062.
[24] Osada, R.; Funkhouser, T.; Chazelle, B.; Dobkin, D.: Shape distributions, ACM Transactions on Graphics, 21(4), 2002, 807-832.
[25] Pal, P.; Tigga, A.; Kumar, A.: Feature extraction from large cad databases using genetic algorithm, Computer Aided Design, 37(5), 2005, 545-558.
[26] Rusinkiewicz, S.; Levoy, M.: Efficient variants of the ICP algorithm, 3DIM, 2001, 145-152.
[27] Smith, S. D. G.; Escobedo, R.; Anderson, M.; Caudell, T. P.: A deployed engineering design retrieval system using neural networks, IEEE Transactions on Neural Networks, 8(4), 1997, 847-851.
[28] Thompson, W. B.; Owen, J.; de St. Germain, H.; Stark Jr., S.; Henderson, T.: Feature-based reverse engineering of mechanical parts, IEEE Transactions on Robotics and Automation, 12(1), 1999, 57-66.
[29] Veltkamp, R. C.; Tangelder, J. W.: A survey of content based 3d shape retrieval methods, in: Shape Modeling and Applications, 2004.
[30] Yamauchi, H.; Lee, S.; Lee, Y.; Ohtake, Y.; Belyaev, A.; Seidel, H.-P.: Feature Sensitive Mesh Segmentation with Mean Shift, Shape Modeling and Applications, 2005