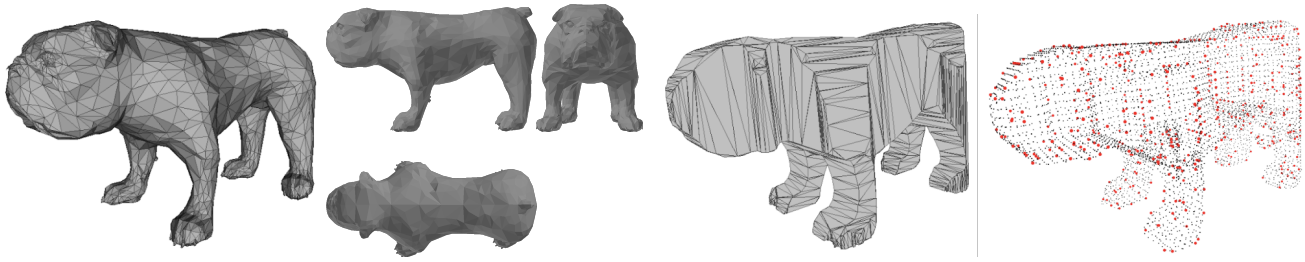
Searching Document Repositories using 3D Model Reconstruction

Cristopher Flagg
Georgetown University
chf38@georgetown.edu

Ophir Frieder
Georgetown University
ophir@ir.cs.georgetown.edu



(a) Original representation of 3D model **(b) Document representation of model in 3-view** **(c) Model reconstructed from document 3-view** **(d) Final point cloud with FPFH key points**

Figure 1: Representation of a 3D model as a document and the subsequent model reconstruction.

ABSTRACT

A common representation of a three dimensional object is a multi-view collection of two dimensional images showing the object from multiple angles. This technique is often used with document repositories such as collections of engineering drawings and governmental repositories of design patents and 3D trademarks. It is rare for the original physical artifact to be available. When the original physical artifact is modeled as a set of images, the resulting multi-view collection of images may be indexed and retrieved using traditional image retrieval techniques. Consequently, massive repositories of multi-view collections exist. While these repositories are in use and easy to construct, the conversion of a physical object into multi-view images results in a degraded representation of both the original three dimensional artifact and the resulting document repository. We propose an alternative approach where the archived multi-view representation of the physical artifact is used to reconstruct the 3D model, and the reconstructed model is used for retrieval against a database of 3D models. We demonstrate that document retrieval using the reconstructed 3D model achieves higher accuracy than document retrieval using a document image against a collection of degraded multi-view images. The Princeton Shape Benchmark 3D model database and the ShapeNet Core 3D model database are used as ground truth for the 3D image collection. Traditional indexing and retrieval is simulated using the multi-view images generated from the 3D models. A more accurate 3D model search is then considered using a reconstruction of the original 3D models from the multi-view archive, and this model is searched against the 3D model database.

DocEng '19, September 23-26, 2019, Berlin, Germany

ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

CCS CONCEPTS

• **Information systems** → **Document representation**; *Content analysis and feature selection*; • **Computing methodologies** → **3D imaging**; *Shape representations*; • **Applied computing** → **Document searching**; • **Social and professional topics** → Patents.

KEYWORDS

Document Repository, 3D Modeling, Model Reconstruction, Point Cloud, SIFT, Fast Point Feature Histograms, Patent Documents

ACM Reference Format:
Cristopher Flagg and Ophir Frieder. 2019. Searching Document Repositories using 3D Model Reconstruction. In *DocEng '19: The 19th ACM Symposium on Document Engineering (DocEng 2019), September 23, 2019 to September 26, 2019, Berlin, Germany.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Document repositories that rely on physically reproducible copies of documents, such as paper and pdf representations, make concessions when attempting to describe physical objects. These object descriptions are comprised of multiple representative images drawn to show the likeness of the object from different viewpoints.

Rather than rely on traditional multi-view document representations of the original object, we reconstruct the object as a 3D model using only the multiple viewpoint images in the original document collection. The resulting reconstructed model provides a new representation of the object that is closer to the original model. Consequently, it is possible to more accurately search the document repository using reconstructed model than it is to search the individual images in the document repository directly.

This approach may be applied to many governmental databases of 3D objects which describe Patents, Industrial Designs, and Trademarks as a set of images. These documents contain a textual description of the object and a set of representative images of the object. The United States Patent and Trademark Office (USPTO) issues design patents in this format. For a design patent "[t]he drawings or photographs should contain a sufficient number of views to completely disclose the appearance of the claimed design, i.e., front, rear, right and left sides, top and bottom." [18]

The method proscribed by the USPTO for searching for design patent documents[17] is to identify the proper classification for the invention and then manually review all of the documents assigned to that classification. This method is both time consuming and prone to error since documents may not be assigned to the most obvious classification.

Standard image feature techniques, such as Scale Invariant Feature Transformation (SIFT), transform an image into a list of important features. While these image features capture some details of the model from different vantage points, the image representations contain only information within the silhouette of the image. Any surface variation cues, such as contours or shape information, are represented by line patterns, shading, and hatching marks. The placement of these indicators is in some sense arbitrary and may vary in pattern and spacing. The model may have artistic shading and patterns applied to the model surface. No differentiation is made between details that are visual elements of the model and surface variation cues.

Additionally, the collection of images fails to represent how these images interrelate. This is evident in SIFT's inability to generalize the rotation of an object within an image beyond 20 degrees[15]. Given that most objects are represented as a set of six images, this limits SIFT's ability to recognize details that cross the transition from one silhouette to the next. Details such as surface contours are lost in the transition from one silhouette to another. No information is available in the image sets to relate features or contours across the image silhouette's boundaries.

The external boundary of the silhouette presented in the document representation provides a definitive boundary between the area contained within the model and the area outside of the model. None of the visual ambiguity of surface details are present since the silhouette provides a binary model/not model distinction. By reconstructing the object from a multi-view collection of the external boundary of the image silhouettes, surface details and contours may be generated that provide enough detail for 3D model retrieval and reconstruction of a 3D model database. These 3D methods of comparison provide a better assessment of similarity than image retrieval methods applied directly to the multi-view images.

We validate the following hypotheses:

H1: The reconstructed model is sufficiently similar to the original model to allow retrieval.

H2: The retrieval of the reconstructed 3D models provides a more accurate search than standard image based retrieval methods.

By validating these hypotheses, we demonstrate at least an initial step towards improving the current design patent search approach and a better utilization of existing document repositories.

2 RELATED WORK

Our effort involves the reconstruction of 3D models from images. We also involve 3D retrieval techniques applied to reconstructed models to existing multi-view image retrieval techniques. The following papers are representative work in these areas.

2.1 Reconstruction of 3D from images

A separate body of work focused on reconstruction of 3D models from images. These methods seek a 3D reconstruction of an object through a combination of geometric matching and semantic/functional analysis of each view[1]. This is typically taken from vertex and face reconstruction derived from multiple engineering drawings. Similar approaches [8, 23] look to combine features from the image with extruded volumes estimated from the drawings. These methods use hidden lines and face reconstruction techniques to facilitate final construction of the objects.

Idesawa [10] uses the outer profiles of isometric drawing objects to extrude the parts before further refining the object. These views are then intersected to produce a final volume. This paper is focused primarily on generating polyhedra where the CAD drawings contain no additional surface detail.

Tanaka [25] attempts to produce 3D reconstructions from 2D assembly drawings, rather than renderings of the model from different images. Silhouettes are used to make the wireframe models of the 3D assemblies. All 2D vertices and edges that can exist as silhouettes are drawn in the 2D drawings and are used to create simple shapes that may be combined to create the final object. The method requires assembly drawings in which both individual parts and part sizes are annotated to disambiguate the final 3D design.

Cao [6] presents a method of transforming an isometric view of a planar object into a 3D object by inferring hidden topological structures. Each hidden face is assumed to correspond to a visible face and creates objects with symmetry. Multiple possible hidden structures for a shape are explored, and the shape with the minimum standard deviation of all angles is considered to be the best candidate for a 3D construction. While reconstruction may be achieved form a single drawing, the method does not apply to figures where the hidden faces and visible faces do no share symmetry.

2.2 2d retrieval and patent CBIR systems

Representing a physical object as a collection of images was an acceptable format when the total number of documents was relatively small and could be reviewed by manual inspection. As the number of documents increases the ability to manually retrieve relevant documents becomes more difficult. Automated methods attempt to search these documents[5] by first isolating the representative images in the documents and then applying image retrieval techniques to create a set of searchable features for each document. These features are constructed using standard techniques (such as SIFT[15], SURF[4], and Fourier Transform[12]). Given this technology, retrieval of physical objects is reduced to image retrieval provided the images adhere to the same standards. It is not possible to search across collections where the image submission requirements differ since consistent feature vectors are not possible.

Zeng [27] provides a synthesis design patent image retrieval method based on shape and color features. These moment invariant

features are indexed and query images are retrieved from the collection. The collection is a set of color images taken from fixed view under constant lighting conditions and does not address rotation or view invariant features when comparing design similarity.

While noting "design patent verification based on manual comparison is too labor-intensive, time-consuming and subjective" Zhu [29] uses Block-wise Dense SIFT (Block-DSIFT), Pyramid Histograms of Orientation Gradients (PHOG), and GIST as image features. Features are extracted from the representative design images and clustered using K-Means and finally combined into an aggregate feature for retrieval. The paper requires all designs images to be taken from a consistent view and does not address scaling or rotation invariant features.

Lee [12] searches 3D trademarks comprising a collection of images. Discrete Fourier transform (DFT) is used create the image feature for retrieval. The paper addresses rotation of the images by creating an additional set of rotated database images for each original image resulting in a much larger database.

A trademark content based image retrieval system was announced by the World Intellectual Property Organization[16] to provide image searching for over 4 million images within the database. The site allows retrieval to be filtered based on the shape, color, and texture of a sample image provided by the user. No details are given about the underlying algorithm.

An interesting variation on the standard image retrieval techniques is Su's use of 3D models to generate a multi-view image collection [24]. This work takes a 3D model and uses twelve generated multi-view images as input to a CNN to create a feature vector for the 3D model. In effect, the feature vector used for retrieval is a combination of multiple image feature vectors. If the database and query images are not constructed from a consistent set of views this method may not be applied.

2.3 3D model retrieval

Knopp [11] describes the extension of SURF to be used in the context of 3D shapes. The 3D model is voxelized and the 3D SURF descriptors are generated therefrom. The 3D SURF descriptors may used as features for retrieval or combined in an aggregate feature vector. This method is similar to the use of Fast Point Feature Histograms described in this paper.

Li [14] and Bai[3] propose 3D shape retrieval methods that deconstruct a model into a set of multi-view images to be fed to a CNN to create a feature vector. These method deconstruct the 3D model into a set of depth images which are then used to generate the feature vector. These methods require the original model in order to generate the depth imagesand do not apply to existing image document repositories.

Furuya[9] deals directly with 3D models by sampling a set of Rotation Normalized Grids (RNG) features which are then fed into a CNN for refinement and final classification.

Wang[26] uses the Princeton Shape Benchmark data set and free form sketches collected from the Mechanical Turk. A Siamese CNN is fed both the sketch and a model from either a matching or different classification. The resulting CNN takes a sketch as input and generates a feature vector which is used to match the sketch to 3D models from the collection.

With the exception of Knopp, the above methods use CNNs to take in images or models and generate a final feature vector. Since this paper focuses on both the conversion of 2D images into 3D models as well as searching of 3D models, our focus was to validate retrieval using the reconstructed models. The choice to use direct feature generation over neural network feature generation was made to established a baseline for 2D and 3D retrieval methods from document repositories. Once this baseline is established, future optimization using neural networks is expected to provide stronger support for the methods described in this paper.

3 DATA SETS

3.1 Princeton Shape Benchmark

The Princeton Shape Benchmark[22] provides a repository of 3D models and software tools for evaluating shape-based retrieval and analysis algorithms. The Benchmark contains a database of 1,814 3D polygonal models collected from the Internet. The database is divided into a training set of 907 models and a testing set of 907 models. The data set divides the models according to several categorization schemes. The most specific categorization has 53 categories while the broad categorization has 7 high level categories. There is an additional coarse classification which provides a binary split between man made and natural objects. The data set is available at http://shape.cs.princeton.edu/benchmark/.

3.2 ShapeNet Core

ShapeNet Core[7] is a large-scale data set of 3D shapes collected by researchers at Princeton University, Stanford University, and the Toyota Technological Institute at Chicago (TTIC). There are around 51,300 unique 3D models organized using the WordNet Hierarchy. The data set has 35,765 training models, 5,519 validation models, and 10,266 testing models. ShapeNet Core covers 55 synset common object categories (e.g., airplanes) and 205 refined sub-synset subcategories (e.g., fighter jet). The full title of the sub-synset category n03335030 is identified using the WordNet hierarchy giving the title "Fighter, fighter aircraft, attack aircraft" This data set was used to benchmark 3D tasks such as 3D model segmentation, 3D shape retrieval, and 3D model reconstruction from single images. The data set is available at https://shapenet.cs.stanford.edu/shrec16/.

4 EXPERIMENTAL METHODS

An issue in analyzing 3D model retrieval using reconstructed images is identifying a ground truth from which to base the retrieval analysis. Many image retrieval databases exist but lack the corresponding 3D models. To resolve this, we use a 3D model database and synthetically create the multi-view images therefrom. This provides the ground truth for the models and an original model against which to compare the reconstructed model.

The Princeton Shape Benchmark and ShapeNet Core benchmarks are used to provide the base models for reconstruction. To simulate a 2D document repository and compare retrieval accuracy of the image based retrieval against the 3D model based retrieval, the data sets are first degraded into multi-view sets of images. These images are used in the image based retrieval analysis. The images are then used to reconstruct the 3D models for comparison against the actual models in the database.

4.1 Converting ground truth to 6-View

The multi-view set of images is constructed from data sets by first scaling the ground truth 3D models to one unit length along the principle axis, then centering the object within a one unit cube. This provides a uniform scaling for objects that may have been created at different scales and sizes. Since the scaling is only with respect to the principle axis there is no distortion of the resulting model. Virtual cameras are placed in a position to view each face of the unit cube, and images are generated from each camera creating a 6-view representation of the model.

The images generated from the ground truth 3D model provide a set of image data that closely represent the document format used in government repositories. The categorization and meta-data from the original 3D model is applied to each of these generated images to create the final annotated 6-view image data set.

4.2 6-View as the basis for image retrieval

Given a set of multi-view images, such as those of the Stanford University Bunny in Figure 2, the first step in retrieval is to construct a feature vector representation of each image. This serves as a primary point of analysis when comparing the similarity of multiple images. SIFT is a common tool to accomplish this goal.

SIFT is an algorithm to create Scale Invariant Feature Transformations that identifies key points in an image using corners and color gradients. A 16x16 region around the key point is divided into 4x4 subregions, and an orientation histogram is created. The collection of orientation histograms from the subregions is the basis for the 128 bin feature vector for that SIFT key point. An average of 1,300 SIFT features are generated per model. Because retrieval focuses on matching full images and not locating one image within another image, there is no need to restrict feature matching to only the features with a valid projection from one image to another. The full set of features for each image may be considered.

The first step is to construct a codebook[19] of SIFT features. Using the training split for the data set, the SIFT features are grouped into k different clusters using the standard k-means algorithm. Each cluster represents a grouping of similar SIFT features, and the cluster center is an entry into the codebook. All of the members of a cluster are more similar to each other than they are to SIFT features in other clusters.

Once the k clusters are established, the SIFT features for each individual image are categorized by cluster c and assigned to the codebook. Each feature f_p is assigned to the cluster whose center c_i is the minimum Euclidean distance to the SIFT feature f_p .

$$c_i = \left\{ \mathrm{f}_p : \left\| \mathrm{f}_p - m_i \right\|^2 \leq \left\| \mathrm{f}_p - m_j \right\|^2 \ \forall j, 1 \leq j \leq k \right\},$$

The count of features f_p assigned to the various cluster centers c_i may be viewed as a histogram of k bins containing the distribution of the SIFT features f_p over the cluster. Since each image may have a different numbers of features, the cluster histogram is normalized such that the sum of the count of all histogram bins is equal to one. The normalized cluster histogram may now be used as a feature vector to describe each image. Similar images have similar normalized features clusters. Dissimilar images have different SIFT features and thus different normalized cluster distributions. For the six images generated for a given model the six cluster histograms are combined before normalization resulting in a single feature vector for the model encompassing all six images. All comparisons against other models use the combined cluster histograms.

To retrieve models from the data set, cosine similarity is used to compare the query model's feature vector to the feature vectors of the models in the data set. Models are ranked from smallest cosine distance to largest cosine distance.

4.3 3-View model reconstruction

The method used to reconstruct the 3D models from the 3-view images is the intersection of the silhouettes of three orthogonal faces taken from the multi-view image set. By using the silhouettes, the generated model will not exceed the visual hull boundary of the original object. While there are issues with occlusions and some fine details, the resulting model is sufficient to form a basis for 3D model searching. By using silhouettes for reconstruction, the top/bottom silhouettes are mirror images, as are the front/back and left/right images. Only three orthogonal views are required to reconstruct the model. These three images form the primary faces of the image set and are necessary to reconstruct the model.

The selected primary face silhouettes are linearly extruded into 3D surfaces representing the model as seen from each of the primary faces. Each of these surfaces represents the possible 3D object as seen from this orientation. When the three extruded objects are intersected, a new volume is created that represents the 3D object as visual hull or maximum possible outline of the actual shape.

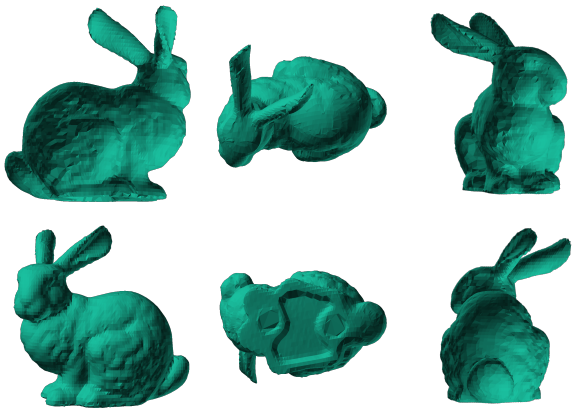
$$\text{Visual Hull} = \text{View}_{\text{front}} \cap \text{View}_{\text{left}} \cap \text{View}_{\text{top}}$$

The resulting reconstructed model is the maximum volume that contains the original model. As shown in Figure 3 (d) the resulting model recreates the contours and interrelations of the primary faces as an approximation of the original shape. Figure 4 shows the deconstruction of a model dinosaur (a) into the 3-views of side (b), front (c), and top (d) views. The 3-view image used to construct the 2D SIFT feature vectors is shown in green and the silhouette used to reconstruct the model is shown in gray. The final reconstructed model (e) is shown in gray.

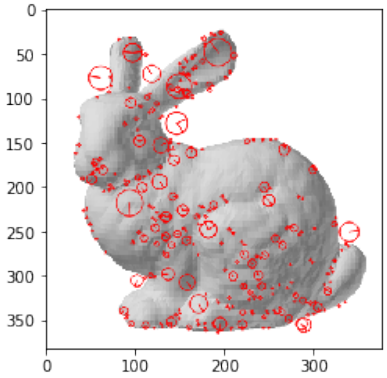
4.4 Reconstructed model to point cloud

The 3D models provided in both data sets are described as collections of planar polygons, typically triangles. Each polygon or triangle is a list of vertices in counter-clockwise order that defines the polygon's outward pointing surface normal. The face of the polygon with the outward facing surface normal is considered the face located on the outside of the surface. When multiple polygons share vertices, this collection of polygons is considered a surface.

It can be difficult to interpret 3D polygon meshes that do not strictly follow this guideline. It is common to define polygons without considering the clockwise/counter-clockwise ordering of the vertices. The result is a surface where the normals of adjacent connected polygons have normals pointing in different directions. Additionally, it is not required that models be composed of a single surface or that adjacent polygons be defined as a single surface. Since both data sets are collections of models taken from the Internet, there is no strong requirement that the models are single surfaces or that the surface normals of the polygons are defined

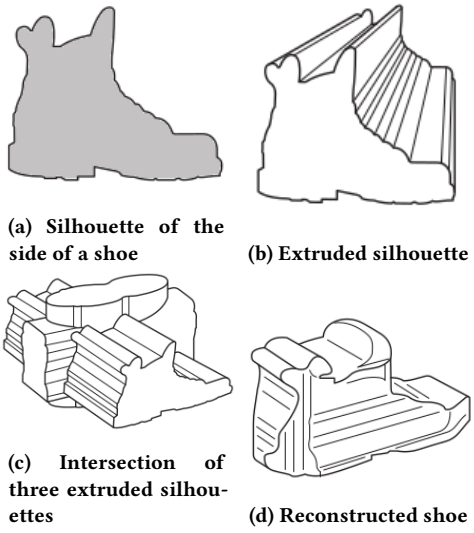


(a) 6-view document representation of Stanford Bunny.



(b) SIFT key points (in red) identified in the front view of Stanford Bunny

Figure 2: Retrieval based on the document representation images of the Stanford Bunny. The SIFT key points are generated for the document representation images and collected as a visual bag-of-words.



(a) Silhouette of the side of a shoe

(b) Extruded silhouette

(c) Intersection of three extruded silhouettes

(d) Reconstructed shoe

Figure 3: Reconstruction of a shoe by extruding and intersecting silhouettes.

in a uniform manner. This adds complexity to the processing of models in mesh format.

Converting the mesh representation of a model into point cloud representation resolves these issues. Point clouds are typically generated using devices such as 3D scanners and LIDAR imaging to create a set of 3D points that represent the visible surface of the object. Rather than representing the model as arbitrarily sized polygons, point clouds describe the model as a set of 3D points arranged across the surface of the model. While these points do not create a complete surface representation of the model, they overcome the discussed difficulties of mesh models.

Using uniform mesh sampling, it is possible to convert a mesh to a point cloud. The polygons of the mesh model are sampled, and three dimensional points are created on the surface of the polygons. The resulting point cloud has three dimensional points distributed in a pattern that corresponds to the mesh surface. Figure 5 shows a guitar in the original polygon mesh format and as a uniformly sampled point cloud.

To create the features used to describe the models, a method called Fast Point Feature Histograms[20] was chosen. Similar to SIFT, features of the region surrounding a query point are gathered and represented as a rotationally invariant descriptor. Points near the query point are identified, and the difference in orientation are treated as point features. A histogram of these features for the k-nearest neighbors is collected and serve as the final feature histogram. Approximately 2800 features are generated per model.

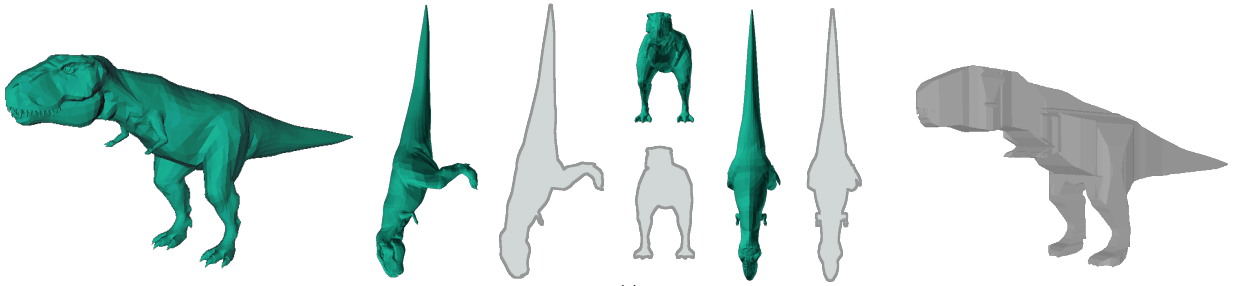
It is important to generate features that will be helpful in model identification. Since many shapes would have multiple points with repeated features, such as all of the points on a plane, it is important to limit the features generated for a model to those that are globally distinct and locally non-repetitive. In Intrinsic Shape Signatures[28], points are chosen based on variance in their eigenvectors to allow for the most variance along the principle components. This identifies the most distinct key points and allows features to be generated for only the most descriptive points. This experiment uses the maximum number of key points generated by the ISS algorithm.

Once the features for the point cloud are identified, a codebook of FPFH features based on the ISS key points is generated and grouped into k different clusters using the k-means algorithm. FPFH features for each model are categorized by cluster c and assigned to the codebook. The count of features f_p assigned to the various cluster centers c_i creates a histogram of k bins containing the distribution of the FPFH features f_p over the cluster. The histogram is normalized and represents the feature vector for the 3D model.

5 EVALUATION

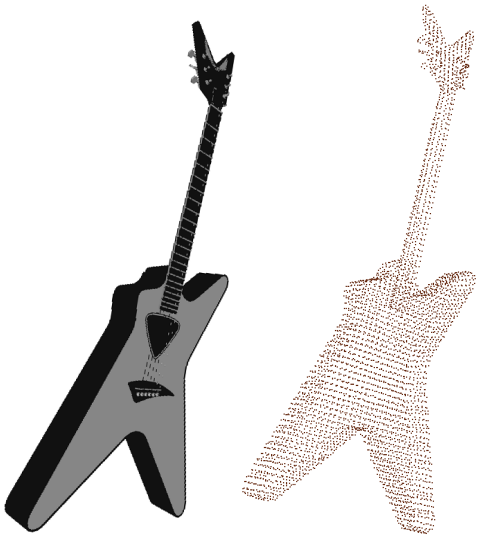
5.1 Timing

Given the above stages outlined for both SIFT and the reconstruction of the original models, Table 1 shows the per model timings



(a) Ground truth model (b) Side View (c) Front View (d) Top View (e) Reconstructed Model

Figure 4: Transformation of ground truth model to reconstructed model using 3-view silhouettes. The green views are images used to construct the SIFT feature vector and the gray views are used to reconstruct the 3D model.



(a) Polygon mesh (b) Point cloud

Figure 5: A polygon mesh uniformly resampled as a point cloud. With respect to the mesh, the gray surfaces have normals facing outward and black surfaces have normals facing inward.

	SIFT	Reconstruction
Create 6-view	0.2055	0.2055
Create SIFT	0.4615	
Image feature vector	0.0474	
Reconstruct Model		0.0260
Create point cloud		0.0461
Create FPFH		1.7679
Model feature vector		3.3675
Generation Time	0.6679	5.4130
Comparison Time:	0.0040	0.0038

Table 1: Timing for the conversion of ground truth models to SIFT and FPFH feature vectors (in seconds) per model for each step of the process.

for each task (in seconds). The processes were run on an 8-core Intel i7 with 64GB ram. The processes did not leverage the graphics card. The generation time includes all transformations from original model, the creation of the SIFT and FPFH features, the average time per model for the k-means clustering of the features, and the construction of the image and model feature vectors. Additionally, the time taken to compare two models is considered. Based on the Princeton Shape Benchmark methodology, the comparison time is the comparison of the query feature vector with the target feature vectors. The comparison time does not include the time required to construct the query feature vector from the query model.

5.2 Similarity of the reconstructed models

The method used for comparing the error between two models is based on the observation that two identical models S and S' will have corresponding points in each model. If the two models are overlaid, the distance from a point in $p \in S$ to the closest point $p' \in S'$ would be zero. If S' were perturbed by altering the position of p', the euclidean distance between p and p' would increase. Therefore the minimum distance between a point $p \in S$ and a model S' would be the minimum distance from p to all points $p' \in S'$ where

$$d(p, S') = min_{p' \in S'} \|p - p'\|$$

Since models S and S' may not contain the same number of points, the error in distance from S to S' is not symmetrical to the error in distance between S' and S. The total error between models S and S' is root mean sum of squared distances from all points in S to the model S' and from all points in S' to the model S:

$$RMSSD_{(S,S')} = \sqrt{\frac{\sum_{p \in S} d(p, S')^2 + \sum_{p' \in S'} d(p', S)^2}{|S| + |S'|}}$$

The RMSSD between two models provides an error metric where small errors represent models with similar shapes and structures

and large errors for models with very different shapes and structures. Since the RMSSD relies on a) the minimum distance between a point and a model and b) both the distance from $p \in S$ to S' and the distance from $p' \in S'$ to S, it provides a symmetrical distance measure [2] where $RMSSD_{(original, reconstructed)} = RMSSD_{(S', S)}$.

For each model, the original model S was compared against the reconstructed model S' using $RMSSD_{(S \in \text{Original}, S' \in \text{Reconstructed})}$, and the average over all original/reconstructed pairs was taken. The RMSSD between the original model and the reconstructed model (Table 3) shows an average error of 0.0369 for the Princeton Shape benchmark and 0.0340 for ShapeNet Core.

5.3 Comparison of models within same classifications

Both the Princeton Shape Benchmark and the ShapeNet Core data sets contain a coarse and a fine level of classification. The most narrow classification covers a specific grouping such as "winged vehicle" in the Princeton Shape Benchmark and "Fighter Jet" within the ShapeNet Core. These models, placed within the same classification, are the most similar models within the data sets. The broad classification provides an umbrella under which several narrow classifications may be collected. The Princeton Shape Benchmark uses "Vehicle" to describe the collection of narrow classifications such as "Winged Vehicles", "Sea Vessels", and "Car". The ShapeNet Core data set uses "Airplanes" to describe a collection of classifications such as "Biplane", "Fighter Jet", "Sea Plane", and "Transport Plane". It should be noted broad ShapeNet Core classification contains 55 categories and the Princeton Shape Benchmark narrow classification contains 53 categories. The broad classification of the Princeton Shape Benchmark is overly broad and is nearly indistinguishable from the full data set.

These numbers provide strong evidence that the reconstructed models are strongly similar to the original models when comparing the average RMSSDs of models within the same class. These classifications group models that should, in theory, be similar and show lower RMSSD when compared to each other than when compared to models in the broader classifications. This hierarchy provides three different levels of granularity with which we can compare the RMSSDs of the models (Table 2).

When all models are compared against each other the similarity is minimal. The average RMSSD for the data set includes the averages of each model against each other model in the data set without regard for classification. The average RMSSD across all models provides an upper bound to the similarity of the models. It should be noted that the standard deviations are closer to 20% greater in the reconstructed models when compared to the standard deviation of the original models. This confirms the reconstructed model has a slightly more general shape than the original model.

By showing the RMSSDs of the reconstructed models track the RMSSDs of the original models for each level of classification, we have established a strong correlation between the original model structure and the reconstructed model structure. In addition to showing strong support for RMSSD as a valid measure of model similarity, this data also details the strong similarity between the original model and the reconstructed model. This suggests that any method of retrieval that provides strong results on the original models will also provide strong results on the reconstructed models.

First, the RMSSD of the original and the reconstructed model pairs is 64% lower than the RMSSD within both the narrowly classified original models and the narrowly classified reconstructed models. Comparing the RMSSD of reconstructed model pairs shown in Table 3 and the RMSSD of the narrow classifications in Table 2 strengthens previous observations about the similarity of the original model to the reconstructed model. If the reconstructed model pairs have an RMSSD greater than the RMSSD of the narrow classification, the reconstruction would alter shape and structure of the original model beyond that of models deemed most similar. If the reconstructed model pairs have an RMSSD on the order of the RMSSD of the narrow classification, then the reconstruction would create a model that was indistinguishable from the other models within the class. If this were not the case, the difference between the RMSSDs of the narrowly classified original models and the narrowly classified reconstructed models would be much greater. The data shows an even stronger relationship than this. Since the reconstructed model pairs have an RMSSD that is 64% smaller than the RMSSD of the narrow classifications, the reconstructed models are much more similar to the original model than the original model is to the rest of the model's class. Additionally, the reconstructed model is twice as distinct from the narrowly classified models as the narrowly classified models are from the data set in general.

5.4 Model retrieval

The feature vector for the document representation / SIFT features and the reconstructed model / FPFH features are defined as a code book constructed using k-means. For this evaluation code books are constructed where the number of clusters is defined by $k = [32, 64, 128, 256, 512, 1024]$. Both sets of feature vectors are created over all k cluster sizes for each reconstructed model. For each of these sets of features,the cosine distance between all of the feature vectors is calculated and the resulting comparisons are ranked from closest (most similar) to furthest (least similar).

The resulting mean average precision across both data sets is shown in Table 4. Retrieval using the reconstructed models consistently performs stronger than retrieval using the document representation. For the Princeton Shape Benchmark, FPFH ($k = 128$) has the top MAP of 0.1967 and SIFT ($k = 64$) has the top MAP of 0.1725. For ShapeNet Core, FPFH ($k = 32$) has the top MAP of 0.3008 and SIFT ($k = 32$) has the top MAP of 0.2570. For both data sets, the reconstructed model retrieval provides more accurate results than retrieval using the document representation.

5.5 Evaluation of Princeton Shape Benchmark models

The following evaluations are provided in the Princeton Shape Benchmark paper[22] for comparison of model shape descriptors:

- **Nearest neighbor**: the percentage of the closest matches that belong to the same class as the query. This statistic provides an indication of how well a nearest neighbor classifier would perform.
- **First-tier and Second-tier**: the percentage of models in the query's class that appear within the top K matches, where

Data Set	Model Type	Narrow Class		Broad Class		All Models	
		RMSSD	std dev	RMSSD	std dev	RMSSD	std dev
Princeton Shape Benchmark	original	0.0859	0.0776	0.1194	0.1335	0.1311	0.1216
	reconstructed	0.0945	0.09815	0.1328	0.1540	0.1418	0.1394
ShapeNet Core (testing set)	original	0.0875	0.0497	0.0892	0.0495	0.1249	0.0876
	reconstructed	0.0939	0.0602	0.0974	0.0618	0.1337	0.1179

Table 2: Root Mean Squared Sum of Distances comparison of models within their own class and with models from the entire data set. Smaller numbers show a higher model similarity.

	RMSSD	std dev
Princeton Shape Benchmark	0.0369	0.0349
ShapeNet Core	0.0340	0.0300

Table 3: Root Mean Sum of Squared Distances Error in reconstruction of models from ground truth.

K depends on the size of the query's class. Specifically, for a class with |C| members, $K = |C| - 1$ for the first tier, and $K = 2 * (|C| - 1)$ for the second tier.

- **E-Measure**: a composite measure of precision and recall that considers only the first 32 retrieved models for every query and calculates the precision and recall over those results.
- **Discounted Cumulative Gain (DCG)**: a statistic that weights correct results near the front of the list more than correct results later in the list under the assumption that a user is less likely to consider elements near the end of the list.
- **Normalized (DCG_a)**: This provides a normalized DCG by comparing the DCG of model a to the average DCG (AverageDCG) for the models compared in the paper, scaling the DCG values to a percentage with relation to the average. Positive/negative normalized DCG scores represent above/below average performance, and higher numbers are better

$$NormalizedDCG_a = \frac{DCG_a}{AverageDCG - 1}$$

Within the statistics provided for the Princeton Shape benchmark evaluation (Table 5), the reconstructed models with FPFH ($k = 128$), provided the strongest results across most measurements and performed better than the strongest document representation with SIFT ($k = 64$) and SIFT ($k = 128$) both providing the highest measurements. The measurements across these representations confirms the results provided in Table 4, showing an advantage to the use of the reconstructed models over the document representation. It should be noted that the highest DCG for the data set was seen with FPFH ($k = 32$) and SIFT ($k = 32$).

When compared to the more specialized feature representations outlined in this paper, the direct SIFT and FPFH feature representations still outranked one third of the original methods. The generation time and the comparison time for both the model reconstruction and document representation were greater than the methods presented in the Princeton Shape benchmark paper. Optimization for these factors was not a consideration in this paper.

5.6 Evaluation of ShapeNet Core models

Of the models used for shape retrieval of 3D models in the SHREC'17 Track[21], 7 were based on deep learning and neural networks while only a single model was based on feature vector matching[13]. Participants were asked to return a ranked list of retrieved models for each query model in a given set, where the target models to be retrieved included the query model itself.

The results for the ShapeNet Core evaluations are provided as a query model followed by an ordered list of matching models with a relevance score, in order from closest match to furthest match. For the evaluations listed in the SHREC'17[21] benchmark, the top 1000 matches were considered. In this paper, only the original data set is considered and the perturbed data set is unused.

The following is a list of the criteria used to evaluate the participants in this track:

- **Precision**: The number of true positives divided by the true positives plus the true negatives.
- **Recall**: the number of true positives divided by the true positives plus the false negatives.
- **F-score**: Precision times recall over by precision plus recall.
- **Mean Average Precision (mAP)**: Mean average precision, average of the precision@k for all results from 1 to N.
- **Normalized Discounted Cumulative Gain (NDCG)**: The NDCG metric uses a graded relevance: 3 for perfect category and subcategory match, 2 for category and subcategory both being same as the category, 1 for correct category and a sibling subcategory, and 0 for no match. This is an attempt at capturing graded relevance between 3D models.
- **Micro and Macro Averaged**: Micro-averaged scores are treated equally across categories and macro-averaged scores give an unweighted average over the entire data set.

The ShapeNet Core evaluation (Table 6) shows the reconstructed models with FPFH ($k = 32$) provides the strongest results across most measurements and performs better than the strongest document representation with SIFT ($k = 32$) and SIFT ($k = 64$) both providing the highest measurements. This evaluation also confirms the results provided in Table 4, showing an advantage to the use of the reconstructed models over the document representation.

The methods detailed in this paper focus on the use of neural networks to directly classify the original model set. While the methods of the ShapeNet Core paper showed a much greater precision and recall than the methods detailed in this paper, the goal of this paper was to compare methods relating to the document representation of 3D models and reconstruction of models from this representation.

		Mean Average Precision					
	Feature Type	$k = 32$	$k = 64$	$k = 128$	$k = 256$	$k = 512$	$k = 1024$
Princeton Shape Benchmark	Document Representation / SIFT	0.1704	**0.1725**	0.1665	0.1510	0.1394	0.1277
	Reconstructed Model / FPFH	0.1838	0.1772	**0.1967**	0.1489	0.1503	0.1501
ShapeNet Core	Document Representation / SIFT	**0.2570**	0.2544	0.2385	0.2268	0.2057	0.2119
	Reconstructed Model / FPFH	**0.3008**	0.2865	0.3001	0.2565	0.2535	0.2454

Table 4: Mean Average Precision of model retrieval using SIFT and FPFH features over both data sets. The features are clustered using a visual bag-of-words code book with cluster sizes of $k = [32, 64, 128, 256, 512, 1024]$. The bold MAP scores show the optimal cluster size for each feature and within each data set.

Shape Descriptor	Storage Size (bytes)	Timing		Discrimination					
		Generate Time (s)	Compare Time (s)	Nearest Neighbor	First Tier	Second Tier	E-Measure	DCG	Normalized DCG_a
SIFT-32	32	0.67	0.003561	32.9%	30.7%	26.7%	13.8%	50.3%	-4.7%
SIFT-64	64	0.67	0.004026	34.5%	30.1%	26.4%	12.8%	49.5%	-4.8%
SIFT-128	128	0.67	0.003932	33.0%	30.7%	27.0%	12.7%	49.4%	-4.8%
SIFT-256	256	0.67	0.004757	30.8%	29.6%	25.2%	11.7%	48.6%	-4.9%
SIFT-512	512	0.67	0.003832	26.6%	24.8%	21.3 %	8.8%	46.4%	-14.1%
SIFT-1024	1024	0.67	0.003852	27.1%	27.4%	23.6%	10.6%	47.7%	-10.2%
FPFH-32	32	5.41	0.004705	34.5%	32.2%	28.3%	14.2%	51.7%	-4.6%
FPFH-64	64	5.41	0.003129	35.2%	31.7%	28.1%	13.0%	50.9%	-4.7%
FPFH-128	128	5.41	0.003818	36.8%	33.0%	28.7%	14.1%	51.2%	-4.6%
FPFH-256	256	5.41	0.003735	30.0%	27.1%	23.7%	8.7%	46.1 %	-14.1%
FPFH-512	512	5.41	0.003729	34.3%	29.4%	24.9%	10.4%	47.9%	-9.6%
FPFH-1024	1024	5.41	0.003751	29.7%	26.1%	22.0%	9.7%	46.6%	-14.0%
LFD	4,700	3.25	0.001300	65.7%	38.0%	48.7%	28.0%	64.3%	21.3%
REXT	17,416	2.22	0.000229	60.2%	32.7%	43.2%	25.4%	60.1%	13.3%
SHD	2,184	1.69	0.000027	55.6%	30.9%	41.1%	24.1%	58.4%	10.2%
GEDT	32,776	1.69	0.000450	60.3%	31.3%	40.7%	23.7%	58.4%	10.1%
EXT	552	1.17	0.000008	54.9%	28.6%	37.9%	21.9%	56.2%	6.0%
SECSHEL	32,776	1.38	0.000451	54.6%	26.7%	35.0%	20.9%	54.5%	2.8%
VOXEL	32,776	1.34	0.000450	54.0%	26.7%	35.3%	20.7%	54.3%	2.4%
SECTORS	552	0.90	0.000014	50.4%	24.9%	33.4%	19.8%	52.9%	-0.3%
CEGI	2,056	0.37	0.000027	42.0%	21.1%	28.7%	17.0%	47.9%	-9.6%
EGI	1,032	0.41	0.000014	37.7%	19.7%	27.7%	16.5%	47.2%	-10.9%
D2	136	1.12	0.000002	31.1%	15.8%	23.5%	13.9%	43.4%	-18.2%
SHELLS	136	0.66	0.000002	22.7%	11.1%	17.3%	10.2%	38.6%	-27.3%

Table 5: Comparisons of the document representation / SIFT features, the reconstructed models / FPFH features, and the features presented in the original Princeton Shape Benchmark paper[22]. For consistency and simplicity of comparison, the abbreviations used in the original paper are maintained here.

6 CONCLUSIONS

The reconstruction of 3D models from a document representation allows for the recreation and close approximation of the original 3D model. The reconstructed model is sufficiently similar to original model to allow for retrieval using point cloud features such as FPFH.

While the document representation lends itself naturally to established 2D retrieval methodologies such as SIFT features, retrieval of the reconstructed 3D models provides stronger results. The reconstruction of the original 3D models not only provides a means for improved document retrieval but also provides an intermediate representation by which documents may be compared to other repositories in different formats. The reconstructed document may serve as a bridge between 2D and 3D repositories.

REFERENCES

[1] Christian Ah-Soon and Karl Tombre. 1995. A step towards reconstruction of 3-D CAD models from engineering drawings. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, Vol. 1. IEEE, 331–334.

[2] Nicolas Aspert, Diego Santa-Cruz, and Touradj Ebrahimi. 2002. Mesh: Measuring errors between surfaces using the hausdorff distance. In *Proceedings. IEEE International Conference on Multimedia and Expo*, Vol. 1. IEEE, 705–708.

[3] Song Bai, Xiang Bai, Zhichao Zhou, Zhaoxiang Zhang, and Longin Jan Latecki. 2016. Gift: A real-time and scalable 3d shape search engine. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5023–5032.

[4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. Surf: Speeded up robust features. In *European conference on computer vision*. Springer, 404–417.

	micro					macro				
Method	P@N	R@N	F1@N	mAP	NDCG@N	P@N	R@N	F1@N	mAP	NDCG@N
SIFT32	0.132	0.221	0.150	0.071	0.204	0.038	0.221	0.052	0.058	0.154
SIFT-64	0.132	0.231	0.151	0.072	0.205	0.040	0.260	0.055	0.062	0.158
SIFT-128	0.118	0.203	0.134	0.057	0.185	0.035	0.221	0.048	0.056	0.151
SIFT-256	0.118	0.200	0.134	0.053	0.178	0.034	0.201	0.047	0.050	0.142
SIFT-512	0.107	0.180	0.122	0.042	0.161	0.031	0.187	0.043	0.046	0.136
SIFT-1024	0.111	0.187	0.126	0.046	0.168	0.032	0.195	0.043	0.045	0.132
FPFH-32	0.146	0.236	0.162	0.087	0.245	0.037	0.242	0.052	0.071	0.174
FPFH-64	0.115	0.204	0.130	0.067	0.206	0.034	0.246	0.049	0.070	0.170
FPFH-128	0.138	0.215	0.153	0.081	0.232	0.035	0.199	0.047	0.064	0.164
FPFH-256	0.107	0.174	0.120	0.051	0.178	0.029	0.181	0.039	0.052	0.146
FPFH-512	0.109	0.191	0.124	0.059	0.188	0.032	0.204	0.044	0.058	0.154
FPFH-1024	0.117	0.199	0.132	0.058	0.190	0.033	0.218	0.046	0.056	0.151
Kanezaki	0.810	0.801	0.798	0.772	0.865	0.602	0.639	0.590	0.583	0.656
Zhou	0.786	0.773	0.767	0.722	0.827	0.592	0.654	0.581	0.575	0.657
Tatsuma	0.765	0.803	0.772	0.749	0.828	0.518	0.601	0.519	0.496	0.559
Furuya	0.818	0.689	0.712	0.663	0.762	0.618	0.533	0.505	0.477	0.563
Thermos	0.743	0.677	0.692	0.622	0.732	0.523	0.494	0.484	0.418	0.502
Deng	0.418	0.717	0.479	0.540	0.654	0.122	0.667	0.166	0.339	0.404
Li	0.535	0.256	0.282	0.199	0.330	0.219	0.409	0.197	0.255	0.377
Mk	0.793	0.211	0.253	0.192	0.277	0.598	0.283	0.258	0.232	0.337
SHREC16-Su	0.770	0.770	0.764	0.735	0.815	0.571	0.625	0.575	0.566	0.640
SHREC16-Bai	0.706	0.695	0.689	0.640	0.765	0.444	0.531	0.454	0.447	0.548

Table 6: Evaluation of the document representation / SIFT features, the reconstructed model / FPFH features, and the features evaluated in the original ShapeNet Core[7] paper. For consistency and simplicity of comparison, the abbreviations used in the original paper are maintained here.

[5] Naeem Bhatti and Allan Hanbury. 2013. Image search in patents: a review. *International journal on document analysis and recognition* 16, 4 (2013), 309–329.

[6] Liangliang Cao, Jianzhuang Liu, and Xiaoou Tang. 2005. 3D object reconstruction from a single 2D line drawing without hidden lines. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, Vol. 1. IEEE, 272–277.

[7] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. 2015. *ShapeNet: An Information-Rich 3D Model Repository.* Technical Report arXiv:1512.03012 [cs.GR]. Stanford University — Princeton University — Toyota Technological Institute at Chicago. https://shapenet.cs.stanford.edu/shrec17/.

[8] Adem Çiçek and Mahmut Gülesın. 2004. Reconstruction of 3D models from 2D orthographic views using solid extrusion and revolution. *Journal of materials processing technology* 152, 3 (2004), 291–298.

[9] Takahiko Furuya and Ryutarou Ohbuchi. 2016. Deep Aggregation of Local 3D Geometric Features for 3D Model Retrieval.. In *BMVC*. 121–1.

[10] Masanori Idesawa. 1973. A system to generate a solid figure from three view. *Bulletin of JSME* 16, 92 (1973), 216–225.

[11] Jan Knopp, Mukta Prasad, Geert Willems, Radu Timofte, and Luc Van Gool. 2010. Hough transform and 3D SURF for robust three dimensional classification. In *European Conference on Computer Vision*. Springer, 589–602.

[12] Chu-Hui Lee and Liang-Hsiu Lai. 2017. Retrieval of 3D Trademark Based on Discrete Fourier Transform. In *International Conference on Mobile and Wireless Technology*. Springer, 620–627.

[13] Bo Li and Henry Johan. 2013. 3D model retrieval using hybrid features and class information. *Multimedia tools and applications* 62, 3 (2013), 821–846.

[14] H Li, T Zhao, N Li, Q Cai, and J Du. 2017. Feature matching of multi-view 3d models based on hash binary encoding. *Neural Network World* 27, 1 (2017), 95.

[15] David G Lowe. 1999. Object recognition from local scale-invariant features. In *iccv*. Ieee, 1150.

[16] World Intellectual Property Office. 2014. WIPO Launches Unique Image-Based Search for Trademarks, Other Brand Information. https://www.wipo.int/pressroom/en/articles/2014/article_0007.html. Media Center, May 2014.

[17] United States Patent and Trademark Office. 2017. https://www.uspto.gov/sites/default/files/documents/7%20Step%20US%20Patent%20Search%20Strategy%20Guide%20%282016%29%20Long%20Version.pdf. June 29th, 2017.

[18] United States Patent and Trademark Office. 2019. Design Patent Application Guide. https://www.uspto.gov/patents-getting-started/patent-basics/types-patent-applications/design-patent-application-guide. February 3rd, 2019.

[19] Guoping Qiu. 2002. Indexing chromatic and achromatic patterns for content-based colour image retrieval. *Pattern Recognition* 35, 8 (2002), 1675–1686.

[20] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. 2009. Fast point feature histograms (FPFH) for 3D registration. In *2009 IEEE International Conference on Robotics and Automation*. IEEE, 3212–3217.

[21] Manolis Savva, Fisher Yu, Hao Su, Asako Kanezaki, Takahiko Furuya, Ryutarou Ohbuchi, Zhichao Zhou, Rui Yu, Song Bai, Xiang Bai, et al. 2017. Large-scale 3D shape retrieval from ShapeNet Core55: SHREC'17 track. In *Proceedings of the Workshop on 3D Object Retrieval*. Eurographics Association, 39–50.

[22] Philip Shilane, Patrick Min, Michael Kazhdan, and Thomas Funkhouser. 2004. The princeton shape benchmark. In *Shape modeling applications, 2004. Proceedings*. IEEE, 167–178.

[23] Simon SP Shum, WS Lau, Matthew Ming-Fai Yuen, and Kai-Ming Yu. 2001. Solid reconstruction from orthographic views using 2-stage extrusion. *Computer-Aided Design* 33, 1 (2001), 91–102.

[24] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. 2015. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE international conference on computer vision*. 945–953.

[25] Masaji Tanaka, Laurence Anthony, Toshiaki Kaneeda, and Junji Hirooka. 2004. A single solution method for converting 2D assembly drawings to 3D part drawings. *Computer-Aided Design* 36, 8 (2004), 723–734.

[26] Fang Wang, Le Kang, and Yi Li. 2015. Sketch-based 3d shape retrieval using convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*. IEEE, 1875–1883.

[27] Zhiyuan Zeng and Wenli Yang. 2012. Design Patent Image Retrieval Based on Shape and Color Features. *JSW* 7, 6 (2012), 1179–1186.

[28] Yu Zhong. 2009. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*. IEEE, 689–696.

[29] Lei Zhu, Hai Jin, Ran Zheng, Qin Zhang, Xia Xie, and Mingrui Guo. 2011. Content-based design patent image retrieval using structured features and multiple feature fusion. In *Image and Graphics (ICIG), 2011 Sixth International Conference on*. IEEE, 969–974.