ACM Symposium on Solid Modeling and Applications (2004)
G. Elber, N. Patrikalakis, P.Brunet (Editors)

Shape Similarity Measurement Using Ray Distances for Mass Customization

Tae Jin Hwang, Kunwoo Lee, Hun Young Oh and Jee Hoon Jeong

School of Mechanical and Aerospace Engineering, Seoul National University, Seoul, Korea
Email: kunwoo@snu.ac.kr

Abstract

Custom-tailored products are defined as products having various sizes and shapes tailored to meet the customer's different tastes or needs. Thus fabrication of custom-tailored products inherently involves inefficiency. To minimize this inefficiency, a new paradigm is proposed in this work. In this paradigm, different parts are grouped into several groups according to their sizes and shapes. For grouping the different parts, similarity measurement algorithm is used. Similarity comparison starts with the determination of the closest pose between two shapes in consideration. The closest pose is derived by comparing the ray distances while one shape is virtually rotated with respect to the other. Shape similarity value and overall similarity value calculated from ray distances are also used for grouping. A prototype system based on the proposed methodology has been implemented and applied to the grouping and machining of the shoe lasts of various shapes and sizes.

Categories and Subject Descriptors (according to ACM CCS): I.5.3 [Pattern Recognition] Similarity Measures; H.3.3 [Information Search and Retrieval] Clustering

1. Introduction

1.1 Motivation

Custom-tailored products are defined as products having various sizes and shapes tailored to meet the customers' different tastes or needs. Thus, the fabrication of custom-tailored products inherently involves inefficiency, although the level may vary depending on the fabrication method. To minimize this inefficiency a new paradigm different from that of the existing mass production is needed. This new paradigm and algorithms for custom-tailored products are proposed in this paper.

The main framework of the proposed method is described below.

1. Classify available sample objects into several groups by measuring differences of sizes and shapes between the objects.
2. Derive a representative shape for each group that contains all data volume in the smallest size.
3. When a new product is ordered, select the most similar representative shape making similarity comparisons of the new product and each representative shape. Use this representative shape as the work-piece.
4. Generate an effective Numerical Control (NC) tool-path for machining only the different portions between the work-piece and the ordered product.
5. Machine the selected work-piece using the NC tool-path derived in step 4. Proper machining conditions are also derived for each region of the work-piece according to the shape difference.

By machining only the different portions of work-piece and new product, the fabrication time can be remarkably shortened. Among all the steps of this approach for mass customization, we focus on step 1 in this paper and describe it in detail.

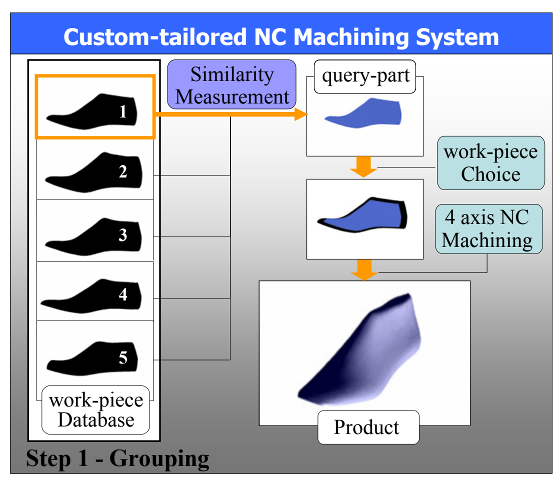

Figure 1: Custom-tailored product system

1.2 Related Work

Many papers regarding similarity comparison have been based on feature information of the feature based models. Elison [ENR97] worked with feature information and McWherter [MPSR01] researched the topological information of models. However, it is more suitable to use a geometric property than to use features for the similarity comparison of models bounded by free-form surfaces.

A geometric approach is similar to the way that humans recognize the difference between two objects bounded by free-form surfaces. When people try to compare two objects, generally they place the two objects in similar positions. Then by observing size differences or curvatures of two objects, they estimate difference in shape. Similarly, for a geometric approach, first the two objects are placed in the most similar position. Then geometric properties (such as volume, curvature etc.) are used to measure the similarities. Thus the similarity comparison problem of a 3-dimensional (3-D) object could be classified into pose estimation and a similarity measurement.

Saupe [Sau00] suggested a pose estimation method using Principal Component Analysis (PCA). According to this method, vectors that can best indicate the shape of the objects are sought first, and optimized positions are decided by allocating the vectors to principal coordinates. Tangelder [TVR03] used the PCA also for pose estimation then made 3D grids that surrounded the objects. By allocating weight values to each cell of grids occupied by the object and by summing the weight values, the similarity was measured.

Shum and Hebert [SHI96] solved pose estimation and similarity measurement problems by triangulating 3-D objects, and comparing curvatures of each node of the triangles. Novotni [NK01] and Klein [KSS99] calculated 2nd inertia momentum of a 3-D object and allocated eigen vectors of inertia tensor matrix to principal coordinates for pose estimation. They used 3-D Distance Field for calculating the similarity. Distance Field is described in detail in Klein and Schilling's work [KS99].

The algorithms introduced above have a common weak point such that it is very difficult to obtain an accurate pose-estimation. Specifically, finding the optimized pose requires a huge amount of computation. If calculations for every pose are avoided, the result of the pose estimation will be poor.

The concept of virtual rotation which is mentioned in Kim at al.'s work [KLH02] is an important key to solving the calculation problem. According to Kim at al., by using a rotating data structure of the geometric property, called virtual rotation, calculation of volume or curvature at every actual rotated pose could be avoided. Thus, the optimized pose could be decided effectively. He made convex hulls for 3-D objects and made a similarity comparison by calculating the volume differences of two convex hulls of the objects at each virtually rotated position. He got a good result, but the unit normal vector set for a pose, which was used as a basic entity for pose estimation, varied after virtual rotation. This required deriving a new unit normal vector set at every virtual rotation.

1.3 Overview of Work

In this paper, the pose estimation and the similarity measurement are achieved by using geometric property. The weak point of the previous works, i.e. the heavy computation for pose estimation, is overcome by using the Ray distance and Kim et al.[KLH02]'s virtual rotation.

First, the reference object and query object are moved such that the centroids of each object accord to the origin of the reference coordinate system. Secondly, unit normal vector sets are generated by casting rays from the origin such that these rays are distributed uniformly around each coordinate axis. Figure 3 illustrates the end points of the rays distributed uniformly around a z-axis. Thirdly, the distances of each ray starting from the origin of coordinates to the object's outer surface are calculated. These distances are defined as Ray Distance to Surface (**RDS**). Normalized values of the **RDS** are defined as Normalized Ray Distance to Surface (**NRDS**). Finally, the optimized poses of the two objects are decided by minimizing total sum of differences between **NRDS** of reference object and that of query object. Once the minimized value is calculated during virtual rotation, the query object is rotated to the optimized position with the angle value derived from the result of virtual rotation.

After pose estimation, two numerical values are derived from **NRDS** and **RDS**. These two values are the Shape Similarity Value (**SSV**) and Overall Similarity Value (**OSV**). **SSV** is a numerical value that expresses the similarity of shapes between the reference object and the query object ignoring the effect of the size differences between two objects. **OSV** is a numerical value that expresses similarity with regard to both the shape and the size. By using these two numerical values, similarity comparison is achieved.

Once **SSV** and **OSV** are calculated, grouping of many objects into several groups is made possible by using these two numerical values. The developed algorithm is applied to the grouping of the shoe last as an essential element to enable mass customization of custom shoes.

2. Pose Estimation of 3D Objects

2.1 Translation

For similarity comparison, the best fitting pose should be derived before similarity measurement. That is, the reference object and query object should be aligned to the position in which two objects look most similar. A 3D object has 6 Degree of Freedom (D.O.F): 3 D.O.F for translation and 3 D.O.F for rotation. By assigning the centroid of the two objects to the origin of the reference coordinate system, 3 D.O.F for translation can be removed. Thus, pose estimation can be achieved by using the remaining 3 D.O.F for rotation.

2.2 Virtual Rotation

2.2.1 Unit Normal Vector Set

For pose estimation of a 3-D object, virtual rotation around 3 coordinate axes of the coordinate system is carried out. For this, data structure containing **RDS** and **NRDS** should be symmetrical along each axis. The term, symmetrical, means that each unit vector should indicate one of the existing directions after virtual rotation. For example, in Figure 2, the left two figures show a unit vector set before virtual rotation, and the right two figures show a unit vector set after virtual rotation. Before virtual rotation, the left two figures have a unit vector set and have data structure indexes indicated by blue dots. After virtual rotation, the right two figures have a unit vector set and have data structure indexes indicated by red dots. However, the upper right figure has two blue dots. That means the directions corresponding to these blue dots do not exist after the rotation. Therefore, if the data structure is non-symmetrical, the virtual rotation cannot be carried out.

Unit normal vector sets are expressed in a spherical coordinate for each axis, and each vector starts from the origin of the coordinate system. End points of these vectors ($x_{m,n}, y_{m,n}, z_{m,n}$) are located on the surface of a unit sphere. Along these unit normal vectors, rays starting from the origin to a point on the object's surface are generated. The lengths of these rays are defined as **RDS**. The index ***m*** and ***n*** indicate the order of unit normal vector in the spherical coordinates and also the location in the data structure which contains **RDS** and **NRDS**. For example, the unit normal vector set for z axis is expressed in the following equations and shown in Figure 3 The unit normal vector sets for x, y, z axes each are shown in Figure 4.

$$x_{m,n} = \cos(n\beta)\cos(m\alpha) \quad y_{m,n} = \cos(n\beta)\sin(m\alpha)$$
$$z_{m,n} = \sin(n\beta) \qquad \cdots\cdots\cdots\cdots (1)$$
$$\left(M = \frac{360}{\alpha},\ N = \frac{180}{\beta},\ 0 \le n \le N,\ 0 \le m \le M-1 \right)$$

In the equation, α and β indicate the spaces between each vector in degrees, and virtual rotation for pose estimation is performed using these values as the rotating angle resolution.

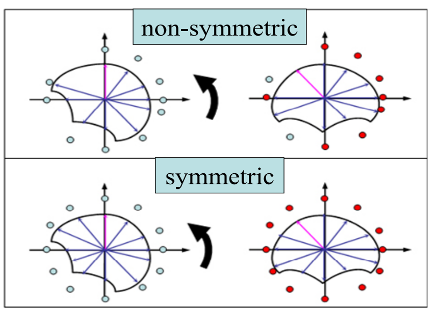

Figure 2: Illustration of the symmetry property of the unit vector set

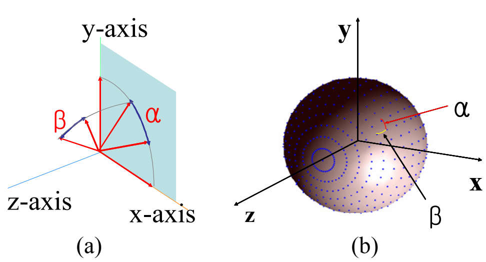

Figure 3: The unit normal vector set for the z axis (a) The unit normal vector (b) The unit normal vector set based on the z axis

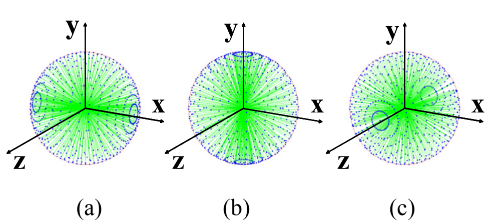

Figure 4: The unit normal vector sets (a) x axis (b) y axis (c) z axis

2.2.2 Ray Distance to Surface (RDS)

By casting rays from the origin in the direction of a unit normal vector in the unit normal vector set, the intersection point of the ray and outer surface of an object is detected. This intersection point for the reference object is denoted by $p^{w}_{m,n}(x_{m,n}, y_{m,n}, z_{m,n})$, and the intersection point for the compared object is denoted by $P^{q}_{m,n}(x_{m,n}, y_{m,n}, z_{m,n})$. Distances from the origin to $p^{w}_{m,n}$ and $p^{q}_{m,n}$ are defined as the Ray Distance to Surface (**RDS**).

$$\mathrm{RDS}_{m,n} = \sqrt{(x_{m,n}-0)^2 + (y_{m,n}-0)^2 + (z_{m,n}-0)^2} \qquad \cdots\cdots\cdots\cdots (2)$$

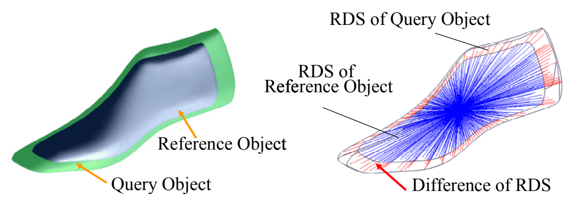

Figure 5: Similarity measurement by the distance of rays

The index ***m*** and ***n*** of $p^{w}_{m,n}$ and $p^{q}_{m,n}$ are also the indices in the unit normal vector set. As stated previously, three unit normal vector sets for three coordinate axes are generated for each object. Thus 6 unit normal vector sets are generated for a pose-estimation.

2.2.3 Normalized Ray Distance to Surface (NRDS)

RDS is a numerical value that indicates the distance between origin and surface, so **RDS** expresses not only the shape but also the size of the object. However, for the pose estimation, the shape of the object is more important than the size of the object. Thus by normalizing the **RDS**, the influence of size should be removed for the pose estimation. The normalization is achieved by calculating the sum of every **RDS** for each axis and dividing **RDS** by the calculated sum. This value is defined as the Normalized Ray Distance to Surface (**NRDS**). As with **RDS**, there are 6 sets of **NRDS** for the standard object and the comparing object.

$$\mathrm{NRDS}_{m,n} = \frac{\mathrm{RDS}_{m,n}}{\sum_{i=0}^{M-1}\sum_{j=0}^{N}\mathrm{RDS}_{i,j}} \qquad \cdots\cdots\cdots\cdots (3)$$

$$\mathrm{NRDS}^{w,x}_{m,n},\ \mathrm{NRDS}^{w,y}_{m,n},\ \mathrm{NRDS}^{w,z}_{m,n},\ \mathrm{NRDS}^{q,x}_{m,n},\ \mathrm{NRDS}^{q,y}_{m,n},\ \mathrm{NRDS}^{q,x}_{m,n}$$

NRDS is a numerical value as sum of the **RDS** is unit. Thus, **NRDS** can be interpreted as a contributing value of whole shape of the object to the direction of unit normal vector.

2.2.4 Virtual Rotation

The **NRDS** for each xyz axis is calculated after translating two objects to the origin of the coordinates. Therefore the optimized pose for comparing object to the reference object should be calculated by virtual rotation. Optimized pose means a position such that the query object aligns to the reference object with minimized sum of difference between **NRDS** of the query object and that of the reference object. As was explained in the overview of work, virtual rotation means an algorithm that decides the optimized pose of the comparing object by changing the index of the data structure. Virtual rotation has the effect as actual rotation, but its calculation cost is less expensive than that of actual rotation.

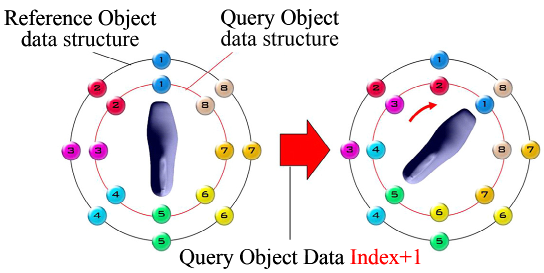

Figure 6: Virtual rotation by shifting memory index

The concept of virtual rotation is follows.

1. At the initial pose of the query object, each sum of the differences between **NRDS** of the reference object and **NRDS** of the query object are calculated for every possible rotation about each of three axes by changing the index of the data structure.
2. In step 1, the index and the corresponding rotation axis of the case having the smallest sum are selected. The query object is rotated actually around the selected axis with the angle value obtained by the product of the index and - α . α is the resolution of the rotation angle about the rotation axis as already used in generating the unit normal vector sets.
3. Step 1 is repeated for the remaining two axes, and the indices and the corresponding rotation axes corresponding to the minimum **NRDS** difference are searched as in Step 2. Then the query object is rotated actually again.
4. Then, the whole process is repeated until the minimum value converges.

For example, the sum of the differences of **NRDS** between the two objects around the x axis is calculated as below. The sum of the differences of **NRDS** between two objects is defined as **SND**.

$$\mathbf{SND}_{x,0} = \sum_{m=0}^{M-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,x} - \mathrm{NRDS}_{m,n}^{q,x}\right| \quad \cdots\cdots (4)$$

If the query object is virtually rotated around the x axis with **kα°**, the **SND** is calculated as in the next equation.

$$\mathbf{SND}_{x,k} = \sum_{m=0}^{M-k-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,x} - \mathrm{NRDS}_{m+k,n}^{q,x}\right| + \sum_{m=M-k}^{M-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,x} - \mathrm{NRDS}_{m-(M-k),n}^{q,x}\right| \quad \cdots\cdots (5)$$

In the upper equation, **k** reflects the effect of rotation with **kα°**, and **x** shows that the rotational axis is **x**. Thus by increasing the value **k** from **1** to **M-1**, every sum of difference of **NRDS** between two objects around the **x** axis can be obtained. For each axis, the **SND** for virtual rotation is as described in the following equation 6.

$$\mathbf{n}(\mathrm{SND}_x)=\mathbf{M}$$

$$\mathrm{SND}_x = \{\mathrm{SND}_{x,0}, \mathrm{SND}_{x,1}, .., \mathrm{SND}_{x,k}, .., \mathrm{SND}_{x,M-2}, \mathrm{SND}_{x,M-1}\}$$
$$(1 \le k \le M-1) \cdots\cdots (6)$$

Similarly, the **SND** for the **y** axis and the **z** axis can be obtained. They are denoted by $\mathrm{SND}_{y,k}$ and $\mathrm{SND}_{z,k}$.

$$\mathbf{SND}_{y,0} = \sum_{m=0}^{M-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,y} - \mathrm{NRDS}_{m,n}^{q,y}\right|$$
$$\mathbf{SND}_{y,k} = \sum_{m=0}^{M-k-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,y} - \mathrm{NRDS}_{m+k,n}^{q,y}\right| + \sum_{m=M-k}^{M-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,y} - \mathrm{NRDS}_{m-(M-k),n}^{q,y}\right|$$
$$\mathbf{SND}_{z,0} = \sum_{m=0}^{M-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,z} - \mathrm{NRDS}_{m,n}^{q,z}\right| \quad \cdots\cdots (7)$$
$$\mathbf{SND}_{z,k} = \sum_{m=0}^{M-k-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,z} - \mathrm{NRDS}_{m+k,n}^{q,z}\right| + \sum_{m=M-k}^{M-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,z} - \mathrm{NRDS}_{m-(M-k),n}^{q,z}\right|$$

The minimum value among $\mathrm{SND}_{x,k}$, $\mathrm{SND}_{y,k}$ and $\mathrm{SND}_{z,k}$ denotes the rotation axis and rotation value for the better pose (where, $1 \le k \le M-1$). Figure 7 illustrates the procedure in which the best pose of the query object is reached by repeatedly applying the virtual rotation.

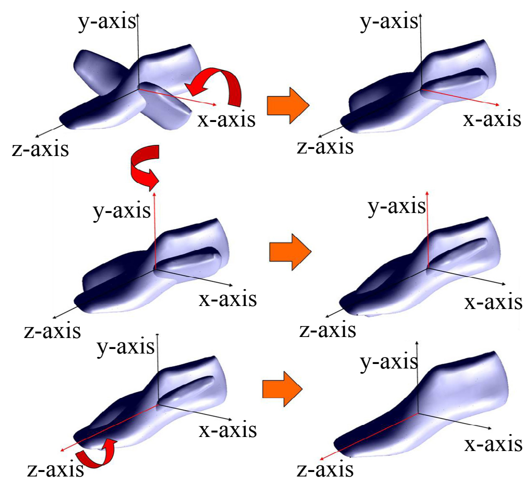

Figure 7: Pose estimation using virtual rotation

3. Similarity Measurement

3.1 Shape Similarity Value (SSV)

SSV is a numerical value that indicates the similarity of the shapes of two objects. This value is concerned with only shapes and ignores the size differences of the two objects. Thus the **SSV** can be obtained by using the **NRDS** of the query object at the best pose as shown next.

$$\mathrm{SSV} = \left[1 - \frac{\sum_{m=0}^{M-1}\sum_{n=0}^{N}\left|\mathrm{NRDS}_{m,n}^{w,z} - \mathrm{NRDS}_{m,n}^{q,z}\right|}{2}\right] \times 100 \quad \cdots\cdots (8)$$

NRDS is the value obtained by dividing each **RDS** by the sum of every **RDS** for each axis. Thus, **NRDS** can be interpreted as the level of the contribution that the whole shape of the object makes in the corresponding unit vector direction. The range of **SSV** defined above can be derived considering the following cases.

1. Two objects have same shape and same size. In this case, every element of **NRDS** for two objects is same, and **SSV** is 100.
2. Two objects have same shape but different size. In this case, every element of **NRDS** for each object is still the same because **SSV** is determined by **NRDS** instead of **RDS**. Thus, **SSV** is 100.
3. Two objects are different in shape and size. According to the equation above, **SSV** can get any number between **0** and **100**. The more similar the shapes of two objects are regardless of sizes, the nearer to **100** the **SSV** is. On the other hand, the more different the shapes of the two objects are, the nearer to value **0.** $0 \le \text{SSV} \le 100$

3.2 Overall Similarity Value (OSV)

For the proposed mass customization system, the size difference of the objects is as important as the similarity of shape. Because the system chooses the work-piece which is most similar to the new product with the removal of the least volume, the size difference of the two objects is also an important value. Therefore the size of objects should be considered for similarity measurement and grouping. **OSV** is calculated by using the **RDS** which contains the size effect.

$$\text{OSV} = \left[1 - \frac{\sum_{m=0}^{M-1} \sum_{n=0}^{N} \left| \text{RDS}_{m,n}^{w,z} - \text{RDS}_{m,n}^{q,z} \right|}{\sum_{m=0}^{M-1} \sum_{n=0}^{N} \left(\text{RDS}_{m,n}^{w,z} + \text{RDS}_{m,n}^{q,z} \right)} \right] \times 100 \quad \text{(9)}$$

In the equation above, by using **RDS** instead of **NRDS**, it is able to reflect the amount of material removal. Thus, by considering the **SSV** and **OSV** simultaneously in grouping, one can arrange the given sample objects into proper groups such that all the objects in each group can be machined from a representative object of the group with a small amount of material removal. Every object has a higher **SSV** and **OSV** between the objects in the same group compared with any one in any other group. The range of **OSV** can be derived by considering the following g cases as was done for **SSV**.

1. Two objects have the same shape and the same size. In this case, every element of **RDS** for the two objects is the same, thus **OSV** is **100**.
2. Two objects have the same shape but a different size. The larger the magnifying power is, the closer **OSV** is to **0**.
3. Two objects are different in shape and size. According to the equation of **OSV**, **OSV** can have any value between **0** and **100**. The more similar the sizes, the nearer to **100** the **OSV** is. On the other hand, the more different the sizes of the two objects are, the nearer to the value is to **0.** $0 \le \text{OSV} \le 100$

4. Results

4.1 Similarity Comparison

The two values, **SSV** and **OSV** are used as a measure of similarity comparison, and applied to the similarity comparison of 10 sample shoe-lasts that are used to produce a shoe. Notice that the shoe last is bounded by free-form surfaces. The **SSV** and the **OSV** for each 10 samples are calculated by using the equations introduced earlier.

Each **SSV** and **OSV** of 10 samples is shown in Figure 8 and Figure 9. The 10 figures illustrated in the left column shows 10 samples with various sizes and shapes. Each sample has an identification number 0 to 9. The numbers in second row have the same identification numbers. For example, if sample number **0** is a reference object, and sample number **4** is a query object the **SSV** for these two objects is **94.65**. Inversely, if the sample number **4** is a reference object, and the sample number **0** is a query object, the **SSV** is **94.65** as expected.

As described previously each value of **SSV** and **OSV** is a number between 0 and 100. Thus the higher each value is, the closer the sizes and shapes of objects are to each reference object.

The **OSV** is derived the same way. Each value is shown Figure 9.

SSV (Shape Similarity Value)										
Data	0	1	2	3	4	5	6	7	8	9
0		96.05	89.54	83.09	94.65	97.25	97.12	98.20	93.04	98.56
1			93.24	92.35	99.10	98.05	96.20	94.21	96.32	96.45
2				98.24	94.32	94.23	85.23	93.45	98.12	88.23
3					93.25	92.13	90.25	91.24	96.25	84.25
4						98.52	91.20	95.30	98.24	93.25
5							96.32	97.40	96.32	97.20
6								94.22	84.25	98.20
7									93.22	96.31
8										92.34
9										

$$\text{SSV} = \left[1 - \frac{\sum_{m=0}^{M-1} \sum_{n=0}^{N} \left| \text{NRDS}_{m,n}^{w,z} - \text{NRDS}_{m,n}^{q,z} \right|}{2} \right] \times 100$$

Figure 8: Similarity Values of sample shoe-lasts

OSV (Overall Similarity Value)										
Data	0	1	2	3	4	5	6	7	8	9
0		94.01	91.30	93.81	96.85	95.63	97.22	96.64	92.68	96.92
1			94.90	93.73	98.74	98.05	90.38	99.79	94.90	94.45
2				96.40	92.02	92.35	91.91	91.23	96.90	90.99
3					91.43	90.87	83.25	89.72	94.71	92.99
4						97.82	96.56	99.54	91.44	98.53
5							93.04	99.30	93.62	94.30
6								95.88	95.95	97.38
7									95.90	95.65
8										89.96
9										

$$\text{OSV} = \left[1 - \frac{\sum_{m=0}^{M-1} \sum_{n=0}^{N} \left| \text{RDS}_{m,n}^{w,z} - \text{RDS}_{m,n}^{q,z} \right|}{\sum_{m=0}^{M-1} \sum_{n=0}^{N} \left(\text{RDS}_{m,n}^{w,z} + \text{RDS}_{m,n}^{q,z} \right)} \right] \times 100$$

Figure 9: Overall Similarity Values of sample shoe-lasts

4.2 Grouping of the Sample objects

When each value of **SSV** and **OSV** is calculated, the grouping process is carried out. The goal of grouping is to classify the samples into several groups with similar shapes and sizes. In this process a measuring number called Grouping Value **GV** is derived from **SSV** and **OSV** as follows.

$$\mathrm{GV} = a \times SSV + (1-a) \times OSV \quad (0 \le a \le 1) \cdots\cdots\cdots\cdots (10)$$

The **GV** is sum of **SSV** and **OSV** with a weight factor ***a***. Since the work-piece derived from the grouping process should have enough volume for each query object, the volume factor, i.e. size, is more important than the shape, and the value of ***a*** should be properly chosen. The user can modify the weight factor ***a*** to classify the sample objects for his/her purpose.

Ten sample objects are grouped into several groups by using the Shape Similarity Value (**SSV**) and Overall Similarity Value (**OSV**). This grouping is performed by a tree structure, and the total number of groups can be controlled according to the user's need. The result of this grouping is shown in Figure 11 when the total number of groups is set to 4. The weight factor for this grouping is **0.5**.

Grouping Value										
Data	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0		95.03	90.42	88.45	95.75	96.44	97.17	97.42	92.86	97.74
1			94.07	93.04	98.92	98.05	93.29	97.00	95.61	95.45
2				97.32	93.17	93.29	88.57	92.34	97.51	89.61
3					92.34	91.50	86.75	90.48	95.48	88.62
4						98.17	93.88	97.42	94.84	95.89
5							94.68	98.35	94.97	95.75
6								95.05	90.10	97.79
7									94.56	95.98
8			**VALUE=(OSV+SSV)/2**							91.15
9										

Figure 10: Grouping Values for grouping of shoe-last

After finding the grouping value, 10 sample objects are classified into several groups. For this grouping, dendograms [AS93] are used.

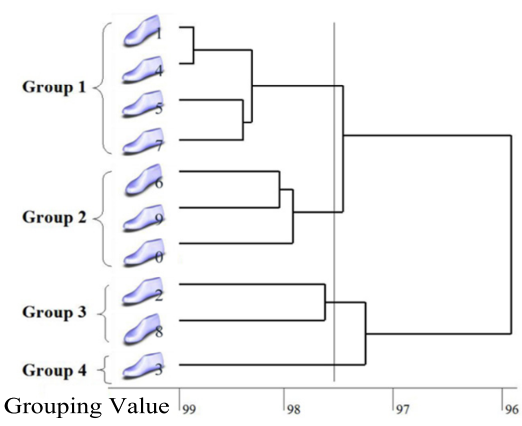

Figure 11: Groups of shoe-last

5. Conclusion

In this paper, a new framework for custom-tailored products and algorithms for this framework are suggested. The proposed method that is based on only geometric properties of the objects for similarity comparison was proven to work well for the objects bounded by free-form surfaces. In addition, an efficient method for pose estimation was developed by using the concept of virtual rotation. The equations which derived **SSV** and **OSV** were also applied successfully to classify the sample shoe-lasts.

The frame work and algorithms of similarity comparison in this work can be applied to the industry of manufacturing custom-tailored products such as the shoe and human wig. This is believed to be an important element of the new paradigm for mass customization.

6. References

[AS93] Askin, R.G., and Standridge, C.R., Modeling and analysis of Manufacturing Systems, Wiley,1993

[ENR97] Elinson, A., Nau, D.S. and Regli, W.C., "Feature based similarity assessment of solid models," Solid Modeling' 97, pp. 297-310, Atlanta, 1997.

[KLH02] Kim, S., Lee, K., and Hwang, T. "A grouping algorithm for custom-tailored products," Journal of Materials Processing Technology, vol. 130-131, pp. 618-625, Elsevier, 2002.

[KS99] Klein, R., Schilling, A., "Fast Distance Field Interpolation for Reconstruction of surface from Contours," Eurographics '99, Short Papers & Demos proceedings, 1999.

[KSS99] Klein, R., Schilling, A., Straβer, W., "Reconstruction and simplification of surfaces from contours," Proceedings of the 7th Pacific Conference on Computer Graphics and Applications, IEEE Computer Society, 1999.

[MPSR01] McWherter, D., Peabody, M., Shokoufandeh, A., Regli, W., "Database Techniques for Archival of Solid Models," Sixth ACM Symposium on Solid Modeling and Application, Michigan, pp. 78-87, 2001.

[NK01] Novotni, M., and Klein, R., "A Geometric Approach to 3D Object Comparison," Proc. Int'l Conf. on Shape Modeling and Applications2001, pp.167-175, Genova, Italy, 2001.

[Sau00] Saupe, D., "Content-Based 3D Model Retrieval," http://www.informatik.uni-leipzig.de/cgip/, 2000.

[SHI96] Shum, H., Hebert, M., and Ikeuchi, K., "On 3D Shape Similarity," Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96), pp 526, IEEE Computer Society, 1996.

[TVR03] Tangelder, Johan W.H., Veltkamp, Remco C., "Polyhedral Model Retrieval Using Weighted Point Sets", Int'l Journal of Image and Graphics, Vol. 3(1), pp.209-229, 2003.