Springer

KSME

Journal of Mechanical Science and Technology 30 (10) (2016) 4705~4716
www.springerlink.com/content/1738-494x(Print)/1976-3824(Online)
DOI 10.1007/s12206-016-0941-8

# A touch-probe path generation method through similarity analysis between the feature vectors in new and old models †

Hyesung Jeon, Jinwon Lee and Jeongsam Yang*

*Department of Industrial Engineering, Ajou University, 206 Worldcup-ro, Suwon 16499, Korea*

(Manuscript Received April 29, 2016; Revised May 23, 2016; Accepted May 23, 2016)

**Abstract**

The On-machine measurement (OMM), which measures a workpiece during or after the machining process in the machining center, has the advantage of measuring the workpiece directly within the work space without moving it. However, the path generation procedure used to determine the measuring sequence and variables for the complex features of a target workpiece has the limitation of requiring time-consuming tasks to generate the measuring points and mostly relies on the proficiency of the on-site engineer. In this study, we propose a touch-probe path generation method using similarity analysis between the feature vectors of three-dimensional (3-D) shapes for the OMM. For the similarity analysis between a new 3-D model and existing 3-D models, we extracted the feature vectors from models that can describe the characteristics of a geometric shape model; then, we applied those feature vectors to a geometric histogram that displays a probability distribution obtained by the similarity analysis algorithm. In addition, we developed a computer-aided inspection planning system that corrects non-applied measuring points that are caused by minute geometry differences between the two models and generates the final touch-probe path.

*Keywords*: Similarity analysis; Feature vector; Path generation; On-machine measurement

## 1. Introduction

The geometry information acquired by a three-dimensional (3-D) measurement instrument is actively utilized for reverse engineering and quality inspection in various industry sectors. A 3-D measurement instrument can be classified as contact-type or non-contact-type depending on whether its sensor contacts the workpiece. Although a contact-type instrument exhibits a slightly lower measuring speed than a non-contact-type that measures the 3-D spatial coordinates of a workpiece using optical technology, such as a camera and laser, it shows a higher degree of measurement precision. This is because a touch-probe is directly in contact with the measuring targets and is less affected by external environmental factors, such as the surrounding illumination and the material on the surfaces of the targets. There are two contact-type measurement methods: the first one uses a Coordinate measuring machine (CMM) that requires additional space for measurement. The second method employs an On-machine measurement (OMM) instrument that directly measures a workpiece within the workspace by changing a tool into a probe on the machine. Although the CMM method achieves higher measurement precision than the OMM, it restricts the size of the workpiece and involves higher initial installation cost. Moreover, the CMM method deteriorates the productivity as it is required to move a workpiece from the machining center to a 3-D measurement instrument after the end of every machining process, such as rough, medium and precision machining. In contrast to the CMM method, the OMM can directly measure a workpiece within a workspace in a way that replaces a tool with a probe in the machining center during the process or after the completion of machining.

While the OMM has the capability of performing compensation machining by immediately correcting various problems that occur during machining through the measurement of a machined workpiece, it has the following limitations. First, the OMM is a time-consuming task and decreases productivity because it is required to assign measuring points on a 3-D shape model, generate a measurement path, and then perform collision inspection although its features are very similar to those of existing models, whose measuring points have already been built. Second, the quality of the measurement is significantly affected by the proficiency of the engineer because the location of the measuring points and paths is established by his or her empirical judgment. To overcome these limitations, in this study, we suggest a touch-probe path generation method that employs similarity analysis between the

*Corresponding author. Tel.: +82 31 219 1879, Fax.: +82 31 219 1610
E-mail address: jyang@ajou.ac.kr
† Recommended by Editor Haedo Jeong

feature vectors of 3-D shapes for the OMM. For the similarity analysis between the new 3-D model and existing 3-D models with already developed measuring points and path, we extracted the feature vectors from models that can represent the characteristics of a geometric shape model; we then applied these feature vectors to a histogram that displays the probability distribution according to the similarity analysis algorithm. After comparing this degree of similarity using the histogram, the existing model that is most similar to the new model is selected. Subsequently, the size of the selected model is scaled to that of the new model and then all measuring points of the similar model are assigned to the new model. If some measuring points are not applied to the new model owing to minute geometry differences between two models, the measuring points must be corrected using measuring-point editing functions provided by our Computer-aided inspection planning (CAIP) system, depending on the feature shapes in which non-applied measuring points are located. Lastly, a touch-probe path is generated by connecting waypoints consisting of measuring points, guide points, and safety points and a simulation for the collision inspection between the probe and the workpiece on the measurement paths is performed.

## 2. Related work

Similarity comparison methods for product models are classified into topology-based, feature-based, and geometry-based, depending on the criteria that are employed to define the similarity for the 3-D shapes. The topology-based comparison method evaluates the similarity by analyzing the connections between topological entities consisting of a geometry model, such as vertices, edges, loops, and faces. While a multi-resolution Reeb graph [1, 2] extracted from the 3-D models compares their topological entities to evaluate the similarity between the 3-D models, a hierarchical skeletal graph [3, 4] composed of three basic skeletal entities (vertex/node, edge and loop) compares the skeleton structures of a voxel model converted from a 3-D model. The topology-based method exhibits good resolution regarding shape variations of complex models but it has the limitation that the similarity of topology does not guarantee that of the whole shape.

The feature-based comparison method assesses the similarity primarily based on machining features. Elinson et al. [5] measured the similarity of solid models by using a graph structure that represents the significant design attributes and important relationships between them. However, this technique was difficult to apply for complicated features as it only targeted some machining features, such as milling and drilling. McWherter et al. [6] constructed a mapping from the boundary representation (B-Rep) of a solid model to a graph-based data structure called a model signature graph and showed how the graph can be used for the similarity assessment of solid models. Meanwhile, Cardone et al. [7] proposed a similarity evaluation algorithm for identifying machined parts in a database that are similar to a given query part; to assess shape similarity, the method utilized reduced feature vectors consisting of machining feature access directions, feature types, feature volumes, feature dimensional tolerances, and feature group cardinality. In addition, Kim et al. [8] suggested a wrap-around algorithm to generate a multi-resolution model by removing or recovering the face set associated with the concave region of a B-Rep model and a smoothing algorithm to simplify the model by removing additional features with relatively small volumes compared to the overall shape. They compared the similarity after changing the complex features of a query model into simplified base features by using those algorithms. Their method can greatly reduce the computational load owing to this simplification but it may exhibit degraded accuracy because major features of the queried model are removed.

The geometry-based comparison method evaluates the similarity by extracting 3-D point cloud data from the geometry information of a shape model and creating a graph representing the geometry shape quantitatively. Osada et al. [9] proposed a method for computing 3-D shape signatures and dissimilarity measures for arbitrary objects described by possibly degenerate 3-D polygonal models. The signature of an object was represented as a shape distribution graph sampled from a shape function describing the global geometric properties of the object. For the similarity measurement, they also experimented with five shape functions: The A3 for measuring the angle between three random points on the surface of a 3-D model, the D1 for measuring the distance between a fixed point and a random point, the D2 for a distance measurement between two random points, the D3 for measuring the area of the triangle formed by three random points, and the D4 for measuring the volume of the tetrahedron defined by four random points. Through this experiment, the D2 shape function was determined to be the most effective in terms of computation time and the discrimination of 3-D shapes. Furthermore, Ip et al. [10] proposed a method for comparing solid models by measuring the distances between two random points on the surface of a 3-D model; the method employed a shape distribution graph created through the D2 shape function. To separate the pairs of points based on the geometric properties of the line connecting them, they classified the distance measurements into three non-intersecting groups: the IN distance where the line connecting the two points lies completely inside the model; the OUT distance where the line connecting the two points lies completely outside the model; the MIXED distance where the line connecting the two points is located both inside and outside the model. Their method showed good results in comparing similarities between volumetric shape models but was difficult to implement for a 3-D polygon soup model that results in overlapping regions on the triangular mesh. Wohlkinger and Vincze [11] introduced a global shape descriptor called an ensemble of shape functions, which is based on three distinct shape functions describing the distance, angle, and area distributions. They evaluated the similarity of 3-D models using the descriptor with the extension of the D2

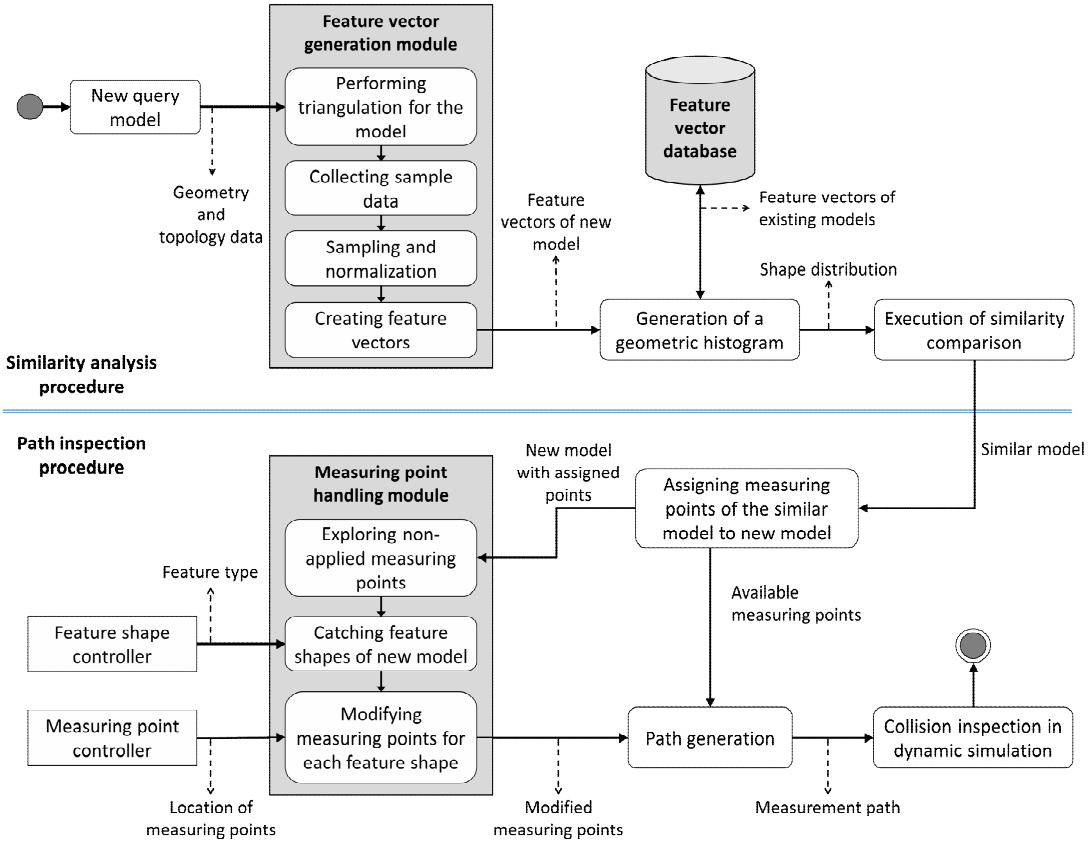



Fig. 1. Overall process for generating a touch-probe path based on similarity analysis.

shape function of Osada et al. [9] and the shape distribution of Ip et al. [10]. In addition, Ohbuchi et al. [12] proposed a shape-similarity search method for 3-D models that used a combination of three feature vectors, such as the inertia moment of the model, the average distance of the surface from the inertial axis, and the variance of the distance of the surface from the axis. However, this method exhibited good similarity recognition rate only when the model was symmetric.

The advantage of the geometry-based comparison method is that by creating a geometric histogram through a shape distribution the similarity comparison is not only easy and simple but also usable universally because it does not require feature and B-Rep information. Therefore, we adopted the geometry-based comparison method. To create the histogram, we used the D2 shape function, which gives robust results compared to other shape functions in measuring the variance of the Euclidean distance on the surface of a 3-D model.

## 3. Similarity analysis of 3-D models

### *3.1 Overview*

The measurement time required for an OMM using a touch-probe installed on the machining center may vary depending on the overall number of measuring points and the skill of engineers. Specifically, even if two workpieces have similar feature shapes, the measuring points and their path should be generated separately; this creates a heavily time-consuming measurement process. To solve this problem, in this study, we propose a method to reuse the measuring points and path of existing 3-D models pre-built in a database for a new model by performing similarity analysis between those models, as shown in Fig. 1. In the similarity-analysis-based CAIP process, the feature vectors are first extracted from all existing models and stored on a feature vector database. If a new model is input into the CAIP system, the geometric histogram that represents the 3-D models as a probability distribution is created by using the feature vectors of the new model and those in the feature vector database. Meanwhile, the probability distribution is sampled from a shape function measuring the global geometric properties of the models. After calculating the Euclidean distance between the feature vectors represented in the histogram, the similarity comparison between the new model and existing models is performed and the existing model with the lowest accumulated distance deviation is selected as the similar model. Lastly, the CAIP system adjusts the scale of the whole size of the similar model to equal that of the new model and then assigns measuring points of the similar model to the new model. If some measuring points are not applied in the new model owing to minute geometry differences between the two models, the measuring points must be corrected using measuring-point editing functions provided by the CAIP system.

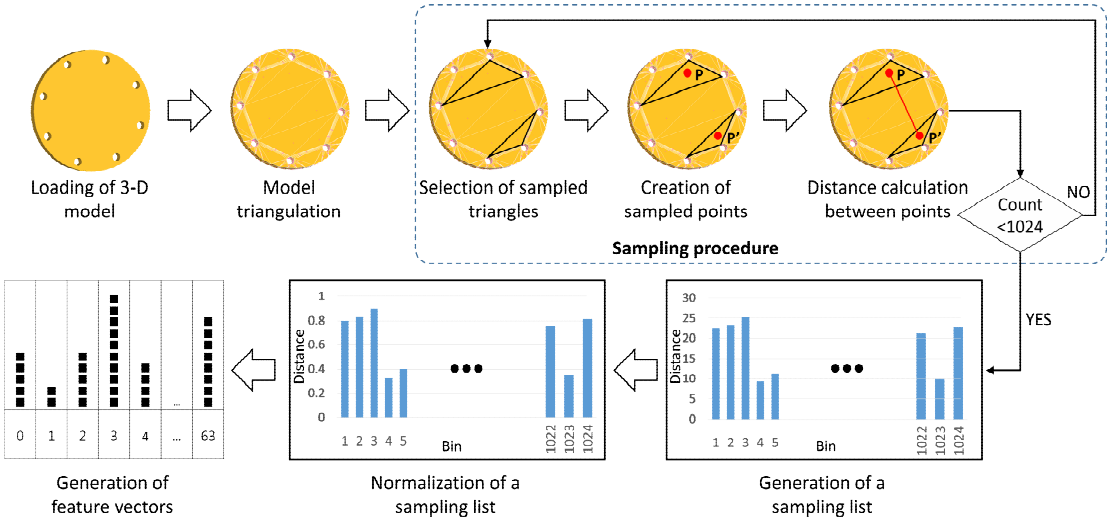



Fig. 2. Creation process of feature vectors from a 3-D model.

Furthermore, we use the similarity analysis algorithm to analyze the similarity between 3-D models quantitatively. This algorithm creates array-typed feature vectors by measuring the Euclidean distance between triangles on the surface of the 3-D model that are represented by triangular meshes; then, it creates a probability distribution histogram using those feature vectors. The accumulated distance deviation between the graph curves of the new model and the existing model that are depicted inside the histogram become a measure of the similarity. This algorithm can perform rapid similarity analysis using the D2 shape function. Additionally, it does not need to match the coordinate systems of the two models because it uses feature vectors that describe the geometric properties of the surface of a shape model to compare the similarity between the two models. Finally, it provides a robust transformation method compared to other similarity comparisons.

### *3.2 Generation of feature vectors*

The similarity is measured by comparing the feature vector values that express the distances between sampled points randomly distributed on the surface of a 3-D model as one-dimensional array. The feature vector generation process includes model triangulation, sampling, generation of a sampling list, and normalization of a sampling list sequentially, as shown in Fig. 2. After converting a 3-D model to triangle meshes, the area of each triangle and the total area of all triangles are calculated to perform the sampling. After the model triangulation is complete, the sampling procedure is performed by selecting some sampled triangles, creating random points inside the sampled triangles, and calculating the distance between points located inside multiple sampled triangles. Simultaneously, the sampling list is generated by using the distances between the sampled points that are obtained by repeating the sampling process 1024 times. In this case, a weighted value is provided to each triangle according to the

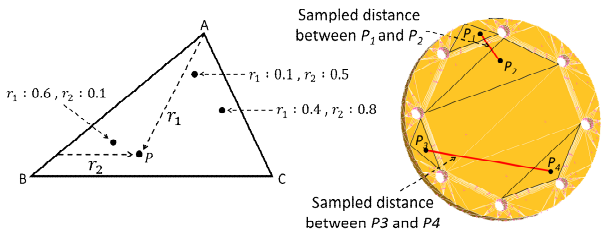



Fig. 3. Selection of a random point within a triangle (left) and creation of the Euclidean distance between two points (right).

ratio of the triangle area to the overall accumulated area calculated during model triangulation; then, it is possible to select an arbitrary sampled triangle. Moreover, as shown in Fig. 3, a random point $P$ is created on the sampled triangle using Eq. (1), suggested by Osada et al. [9], so that all locations inside the sampled triangles can equally contribute to the shape distribution.

$$P = (1-\sqrt{r_1})A + \sqrt{r_1}(1-r_2)B + \sqrt{r_1 r_2}C \quad (1)$$

where $A$, $B$ and $C$ are the location coordinate values (x,y) of the vertices of the triangle, $r_1$ is the ratio between the line $\overline{BC}$ and the vertex $A$ and $r_2$ is the ratio of the inner line segments of the triangle that are parallel to line $\overline{BC}$. The two values, $r_1$ and $r_2$, are generated by using a pseudo-random number generator that produces a non-periodic serial random number sequence. Once point $P_1$ inside a triangle is determined according to $r_1$ and $r_2$ through Eq. (1), point $P_2$ inside the other triangle is produced with the same process. In this case, the Euclidean distance between $P_1$ and $P_2$ on the surface of the 3-D model becomes a sample.

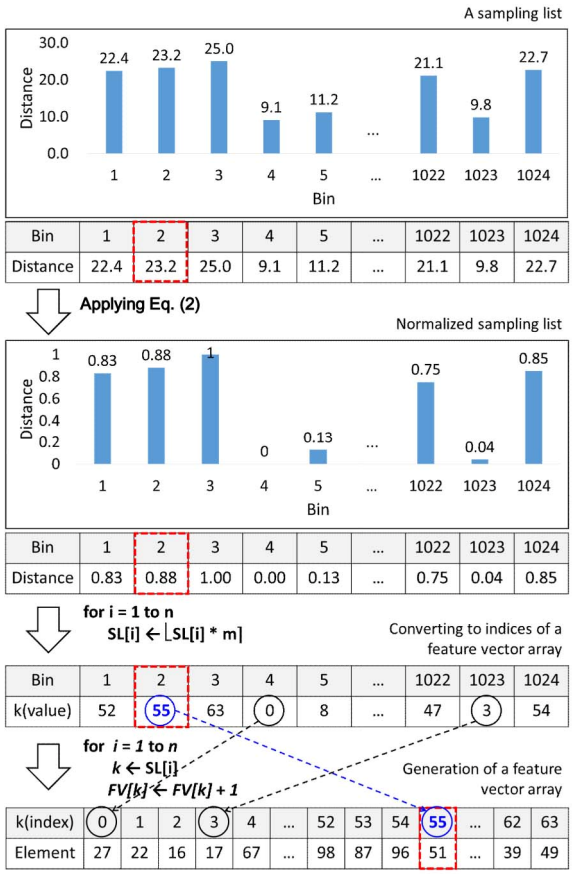
A greater number of sampling times achieves a higher accuracy for finding a similar model and increases the execution time of the sampling. Therefore, we considered 512, 1024 and 2048 sampling times to determine a trade-off for the experi-

ment. With 1024 sampling times, a greater distinctiveness was obtained in the histogram than with 512 times. Although 2048 sampling times achieved better distinctiveness than 1024 times, the former case required greater execution time, which complicated the similarity analysis of 283 existing models in real time. Therefore, we set the sampling times to 1024 and then collected a sampling list. The sampling list was calculated using the Euclidean distances between points. However, if the overall sizes of two models with the same feature shapes are different, their geometric histogram is generated differently depending on the distance range of the sampling list. This might create differences in the similarity analysis. Therefore, we normalized the distance values between points in the range [0, 1] using Eq. (2) so that the similarity analysis could be robust for scale variations.

$$X_{i,0to1} = \frac{X_i - X_{Min}}{X_{\mathrm{Max}} - X_{Min}} \tag{2}$$

where $X_i$ is the i[th] distance value, $X_{Min}$ and $X_{Max}$ are the minimum and maximum values for a sampling list, respectively, and $X_{i,0to1}$ denotes the i[th] normalized distance value with a range of [0, 1]. Therefore, 1024 normalized distance values of the sampling list were transformed to a one-dimensional array of feature vectors with 64 elements through Eq. (2).

Fig. 4 shows the generation process of a feature vector from the sampling list. Fist, the feature vector is created by applying Eq. (2) to each array element. In this case, the integer values *m* and *n* indicate the size of a feature vector and the sampling time, respectively, and the real values *SL[i]* and *FV[k]* are the i[th] element of a normalized sampling list and the k[th] element of a feature vector array, correspondingly. If 64 feature vectors are used, the integer *m* is 63 because the index of a normalized sampling list starts with 0. Meanwhile, the integer *k*, which represents the k[th] index in the array of feature vectors, is rounded to an integer because it is derived from the real number *SL[i]*. The distance 23.2, the second bin number in the sampling list of Fig. 4, changes to 0.88 by Eq. (2) during the normalization process. Finally, the normalized distance value 0.88 is converted into the feature vector index value 55; then, 1 is added to the 55[th] element of the feature vector array. The feature vectors are generated by repeating the same process 1024 times.




| Bin | 1 | 2 | 3 | 4 | 5 | ... | 1022 | 1023 | 1024 |
|---|---|---|---|---|---|---|---|---|---|
| Distance | 22.4 | 23.2 | 25.0 | 9.1 | 11.2 | ... | 21.1 | 9.8 | 22.7 |

| Bin | 1 | 2 | 3 | 4 | 5 | ... | 1022 | 1023 | 1024 |
|---|---|---|---|---|---|---|---|---|---|
| Distance | 0.83 | 0.88 | 1.00 | 0.00 | 0.13 | ... | 0.75 | 0.04 | 0.85 |

| Bin | 1 | 2 | 3 | 4 | 5 | ... | 1022 | 1023 | 1024 |
|---|---|---|---|---|---|---|---|---|---|
| k(value) | 52 | 55 | 63 | 0 | 8 | ... | 47 | 3 | 54 |

| k(index) | 0 | 1 | 2 | 3 | 4 | ... | 52 | 53 | 54 | 55 | ... | 62 | 63 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Element | 27 | 22 | 16 | 17 | 67 | ... | 98 | 87 | 96 | 51 | ... | 39 | 49 |

Fig. 4. Process of generating a feature vector array from a sampling list.

### 3.3 Similarity comparison between feature vectors

For the similarity comparison, we divided the 1024 distance values in a sampling list into 64 sections, creating a one-dimensional array with the feature vectors. The integer value stored in each element of the array represents the number of specific distance values included in the corresponding section. Since the number of the distance values describes the characteristics of the geometric shape, it can determine the similarity between the models through the array of the feature vectors. First, by applying the feature vectors to Eq. (3), a probability distribution is generated from the feature vectors on a geometric histogram. Then, the similarity is measured by comparing the probability values of each bin in the histogram.

$$H_i = \frac{FV_i}{\sum_{i=1}^{n} FV_i} \tag{3}$$

where $H_i$ is the i[th] histogram value, $FV_i$ is the i[th] element value of the feature vector array, and *n* is the size of the feature vector array. The comparison for the histogram is performed using the Minkowski distance metric, which generalizes the distance between two points as shown in Eq. (4).

$$L_p(x,y) = \left(\sum_{i=0}^{n} |x_i - y_i|^p\right)^{1/p} \tag{4}$$

where $L_p(x,y)$ is a measured similarity value, *n* is the size of the bin, $x_i$ is the i[th] value in the histogram of the new model, and $y_i$ is the i[th] value in the histogram of the comparison model. The Minkowski distance is typically used with *p* being 1 or 2; which indicate the Manhattan and Euclidean distance, respectively. In this study, we use the Manhattan distance for rapid calculation.

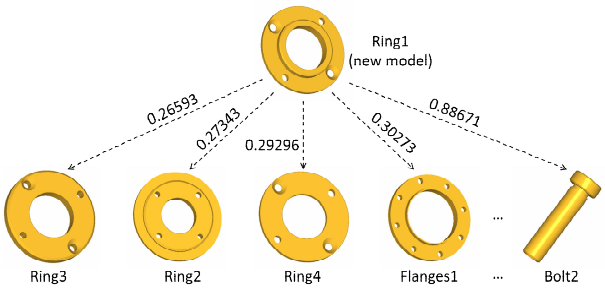
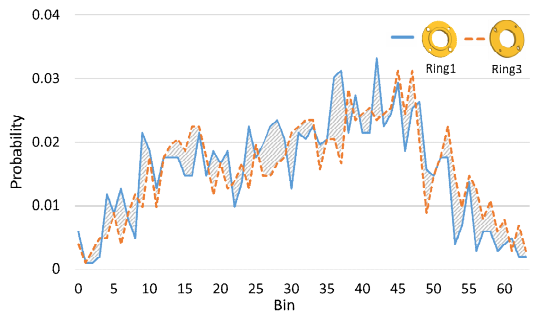
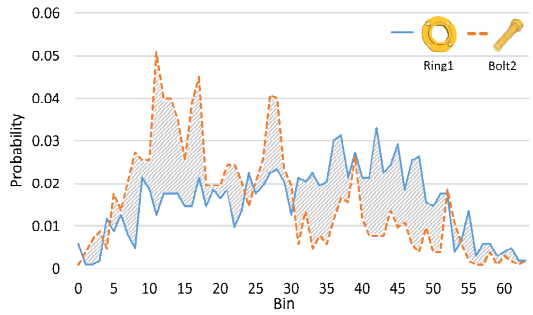



Fig. 5. Example of similarity comparison results for similar models against ring1.




(a) Similarity comparison between rings1 and 3




(b) Similarity comparison between ring1 and bolt2

Fig. 6. Example of geometric histograms with a probability distribution.

Fig. 5 shows the comparison results of similar models against the new query model ring1, organized in descending order of similarity. The similarity was calculated with Eq. (4) and this is the sum of the difference between histogram values. Therefore, a higher similarity between a model and Ring1 is indicated by similar trends and narrower width of the histogram curves of the two models. As shown in Fig. 6, the model that is most similar to ring1 is ring3, corresponding to a similarity measurement of 0.26593 and a similar number of bin values distributed in a specific section of the distance. Accordingly, the least similar model to ring1 is bolt2, its measurement is 0.88671, and the difference between the histogram curves of the two models is wider than that of ring1 and ring3. Therefore, a higher similarity of a model to ring1 corresponds to smaller histogram area difference and a measurement that approaches 0.

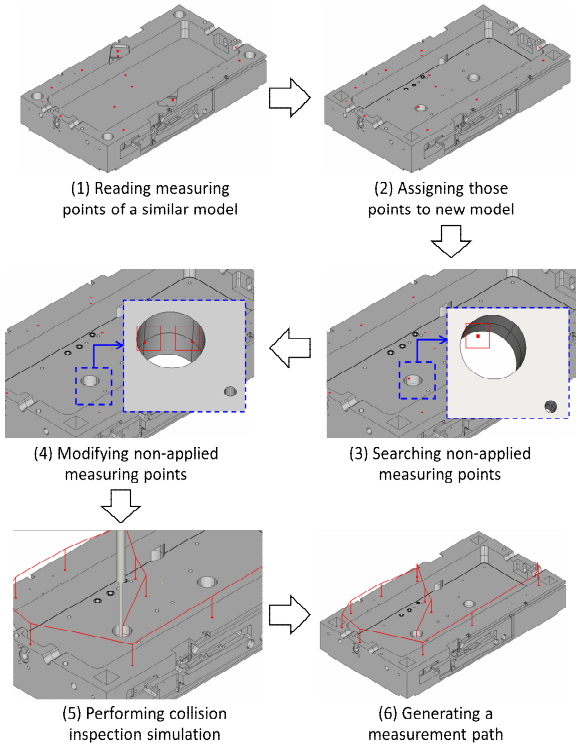



Fig. 7. Process for reusing measuring points of the similar model in the new model.

## 4. Generation of a touch-probe path

### *4.1 Reuse of measuring points of a similar model*

The measurement path of the new model is generated by following six steps, as shown in Fig. 7. First, after selecting a model with higher similarity among existing models using the feature-vector-based similarity analysis, the measuring points of the similar model are transferred to the new model. However, when the measuring points of the similar model are applied to a new model as they are, there might be non-applied measuring points owing to differences in the feature shapes of the two models. The reason for this is the application of the measuring points to unintended locations, which is caused by the difference in scale between the similar model and the new model. This problem can be solved in a way that adjusts the scale of each feature shape or the overall shape of the similar model to that of the new model and then applies the adjusted measuring points to the new model. Another reason may be that the shape of the new model is not perfectly matched with that of the similar model owing to differences in similarity. In this case, it is required to modify the transferred measuring points to match with the size and type of the feature shapes of the locations of the non-applied measuring points. Lastly, once all measuring points are created in the new model, a touch-probe path is generated by connecting these points that performs a dynamic simulation for the verification of the path by collision inspection.

The measuring points of a 3-D model are determined by the

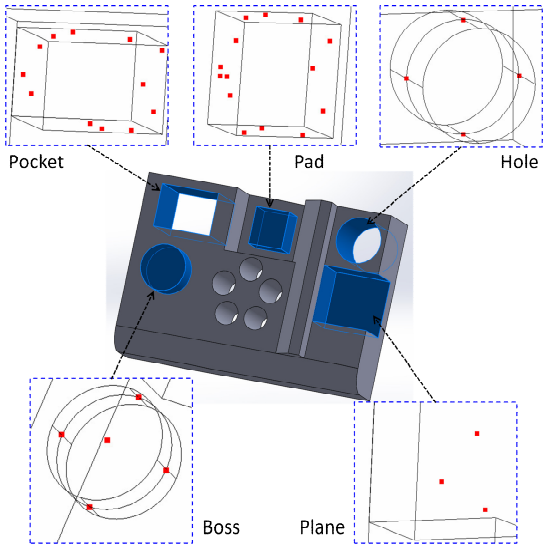



Fig. 8. Measuring point generation method for five feature shapes.

measurement variables depending on the feature shapes of the model. These measurement variables can be obtained by the calculation of the 3-D location coordinates of the measuring points that are created on the feature shapes; examples include the diameter of a hole, the roundness of a circular shape, and the slope of a plane. In this study, we created measuring points for five feature shapes (hole, plane, pad, pocket, and boss), as shown in Fig. 8, that are frequently used in the die/mold machining process. To calculate the measurement variables, the hole feature was assigned to four points at the same height of the internal surface on a cylinder of the feature whereas the plane feature was assigned to three points to verify the surface smoothness. The pad feature, consisting of four planes, was assigned to three points in each plane and two additional points to the bottom and top plane to measure the height. The pocket feature was assigned to three points in each of four planes of the feature. Lastly, the boss feature was assigned to four points at the same height of the internal surface of a cylinder of the feature and one point to each bottom and top plane of the boss to measure the height. Using the location coordinates for these measuring points, it is possible to calculate the measurement variables for each feature shape.

### *4.2 Path generation*

A touch-probe path is generated in the new model from the measuring points specified for each feature shape of the model, in addition to the offset and guide points, to avoid collision between the probe and the workpiece. The probe on the measurement path, as shown in Fig. 9, approaches the workpiece in the direction of the normal vector to each point in the order: guide point, offset point, and measuring point. After contacting the surface of the model, the probe returns by following the reverse order and moves to the next measuring point. The measuring point is the location that is reached when the probe

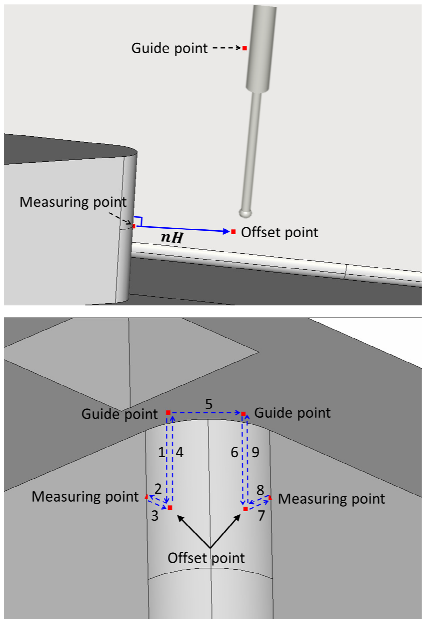



Fig. 9. Method for positioning guide points and offset points on the movement path.

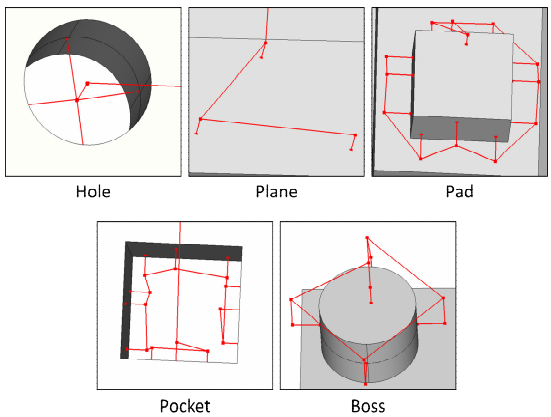
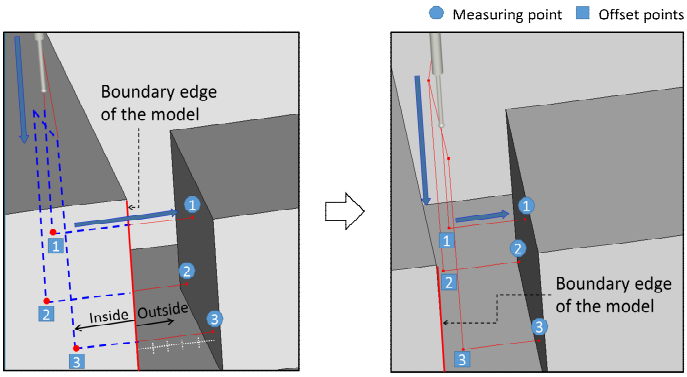


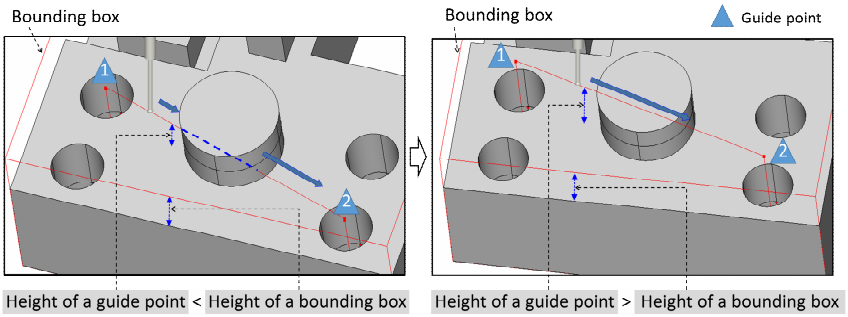

Fig. 10. Probe movement path created on a feature shape.

tip with the touch sensor contacts the surface. The offset point, which is an intermediate point to the measuring point, is a preliminary location before the measurement that corresponds to the height H in the direction of a normal vector n away from the measuring point. The guide point is a location used to avoid collision between the probe and the model when the probe moves to other measuring points. After creating the movement path (1→2→3→4) for one measuring point, the probe moves to another measuring point following the two guide points (4→5). The repetition of this process can complete the overall touch-probe path.

Fig. 10 shows the results of the creation of the movement path of a probe for the five feature shapes. The hole feature, with four measuring points, was assigned to an offset point in the center of the cylinder so that the probe moved only when it entered into the hole and exited after the measurement. In this case, only one safety point was assigned to shorten the meas-




(a) Collision on the entering path (left) and modified path (right)




(b) Collision on the movement path (left) and modified path (right)

Fig. 11. Modification method of offset points and guide points to avoid collision on the entering path or movement path.

urement time. The plane feature, for continuity of the preceding movement path, was created to connect three consecutive measuring points and three consecutive offset points, beginning at the nearest measuring point of the plane from a guide point of the previous feature shape. The pad feature was assigned to a measuring point on the top and bottom of the pad for measuring the height and then to 12 measuring points on the four projected side planes. In this case, an offset point needed to be assigned in the direction of a normal vector at each measuring point. After measuring three points on one side plane of the pad, there could be a risk for collision between the probe and the edges of the pad when the probe moved to the next plane. Therefore, the guide point was created in the same direction as an offset point. The pocket feature created a measurement path by a similar procedure as in the case of the pad. After creating a path to connect the offset points for the four interior planes of the pocket, one guide point was defined only for the first measuring point, in preparation for the moment of entering and exiting the pocket after the measurement. Finally, in the case of the boss feature, measuring points for the projected cylinder surface were created and then offset points were defined in the direction of the normal vector at those measuring points. Moreover, it was required to set a guide point at a higher position than the height of the boss to avoid a collision between the probe and the boss. Finally, after a path for four measuring points on the cylinder surface was created, a measuring point was defined for each of the top and bottom planes of the boss. Therefore, the path of the boss feature consisted of six measuring points, six guide points, and six offset points.

Once the measurement path generation is completed, a collision inspection is required for the entering path and the movement path to prevent the breakage of the probe. As a collision on the entering path occurs when the probe approaches a measuring point, this process assesses whether an offset point or part of the path is located inside the model. If a collision occurs on the entering path, it is necessary to move an offset point in the direction of the corresponding measuring point until it contacts the border of model. Then, the offset point is moved outward by 1/5 of the distance between the border of the model and the measuring point so that the offset point is positioned outside the model, as shown in Fig. 11(a). Furthermore, a collision on the movement path occurs when the probe moves from one guide point to the next, as shown in Fig. 11(b). After solid voxelization of the probe model and the workpiece model is performed for the collision inspection on the movement path, a user evaluates whether the two models overlap on the overall movement path. If any collision has occurred on the movement path, such a collision can be avoided by modifying the location of a guide point. However, if the guide point is set at a higher position than needed to avoid collision, the measurement process becomes inefficient owing to the increased measurement time. Therefore, after calculating a bounding box from the workpiece model, it is recommended to set the guide point at a slightly higher position than that of the contour of the box.

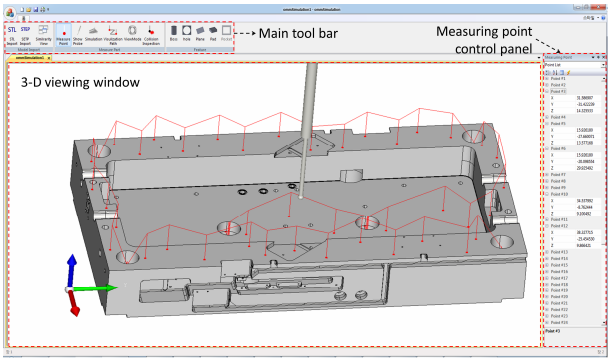



Fig. 12. Configuration screen of our CAIP system.

## 5. System implementation and experiment

### 5.1 Implementation of CAIP system

In this study, we implemented a CAIP system using a Microsoft C++ 7.0 compiler in a Windows 7 operating system environment to verify the proposed method. This system is based on OpenCasCade 6.8, a geometric modeling kernel for geometric calculation and similarity analysis. In addition, this system used Microsoft foundation classes for user interface controls and the OpenGL 3.3 library for geometry visualization. The feature vectors extracted during the similarity analysis were stored using MS Access 2010, a relational database management system, and CAD files saved in STEP or STL format.

This system, as shown in Fig. 12, consists of a main tool bar, a 3-D viewing window, and a measuring point control panel. The main tool bar contains a series of drop-down command menus involving the input and output of the CAD file, a viewport property setting for displaying rendered objects, an interface command for database activation, and a command for editing path points. The path point editing command allows users to set and change the locations of the measuring, offset, and guide points as well as the sequence of these points that is necessary to generate a measurement path through the contextual menu. Moreover, the main tool bar provides interactive functions created by the similarity analysis to dynamically manage the collision inspection on the measurement path.

The 3-D viewing window allows the control of guide points and offset points by moving the probe against non-applied measuring points that are determined after the measuring points of the similar model are applied to the new model. In addition, it is also possible to evaluate the final measurement path on a real-time basis following the movement of the probe. Lastly, the measuring point control panel enables a user to test the location coordinate settings of the guide points, offset points, and measuring points on the 3-D viewing window and fine-tune them by user-input on the view window when mouse control is difficult. Therefore, the dynamic simulation process for the applied measurement path on the 3-D viewing window is connected to the measuring point control panel so that a user can monitor the location coordinates of all points in real time.

### 5.2 Experiment

For the path generation experiment, we applied a 3-D mold base model named DieSet to our CAIP system. The model was designed for manufacturing the middle frame of a mobile phone. The DieSet model, which was saved in STEP format, consists of 83 feature shapes and it was input into the system following the model triangulation process. When a user selects the similarity comparison button, as shown in Fig. 13, the system searches the feature vector database, which contains 283 existing models and their feature vectors, and performs a similarity analysis in comparison with the DieSet model. The analysis result, along with the model name and histogram, is sorted by order of similarity value using the similarity viewer window. When selecting a desired model among the existing models arranged in this window, it is possible to inspect its similarity value against the DieSet model and the location coordinates of the measuring points in a preview pane. The experiment determined that the most similar model to DieSet model was the MoldBase model, whose similarity value was 0.179297. If a user selects the MoldBase model by double clicking, the scale calculated by the bounding box for this model is adjusted to equal that of the DieSet model and then all measuring points of the MoldBase model are transferred into the DieSet model.

Although the measuring points for the MoldBase model were applied to the DieSet model, some measuring points were not correctly positioned on the surface of the DieSet model owing to minute geometry differences between the two models, as indicated by the square in Fig. 14. These non-applied measuring points may occur when they are indented into the model or are separated from the external side of the model. The non-applied measuring point *pt#3* did not exist in the MoldBase model but it corresponds to a location at a newly created hole in the DieSet model. Moreover, two points, *pt#44* and *pt# 45*, were located inside the shape owing to geometry differences of the two models. Furthermore, four points, from *pt#11* to *pt#14*, which were used for measuring the hole in the MoldBase model, remained as non-applied measuring points in the pocket of the DieSet model. These non-applied measuring points were automatically detected, whether they were located on the surface of the model or not, by searching the coordinates using a measuring point control panel provided by the system. The point *pt#3* was modified into four measuring points to be applied to the hole of the DieSet model using the editing functions in the measuring point control panel, and both *pt#44* and *pt#45* were moved on the surface of the model by employing a user-defined axis. Therefore, four points (*pt#11~pt#14*), which were used to measure a hole in the MoldBase model, were changed into 12 points suitable for the pocket in the DieSet model.

Once all measuring points for the DieSet model are set, it is possible to generate the measurement path for the touch-probe by connecting these measuring points. However, because the probe may collide with the model on an entering path or a

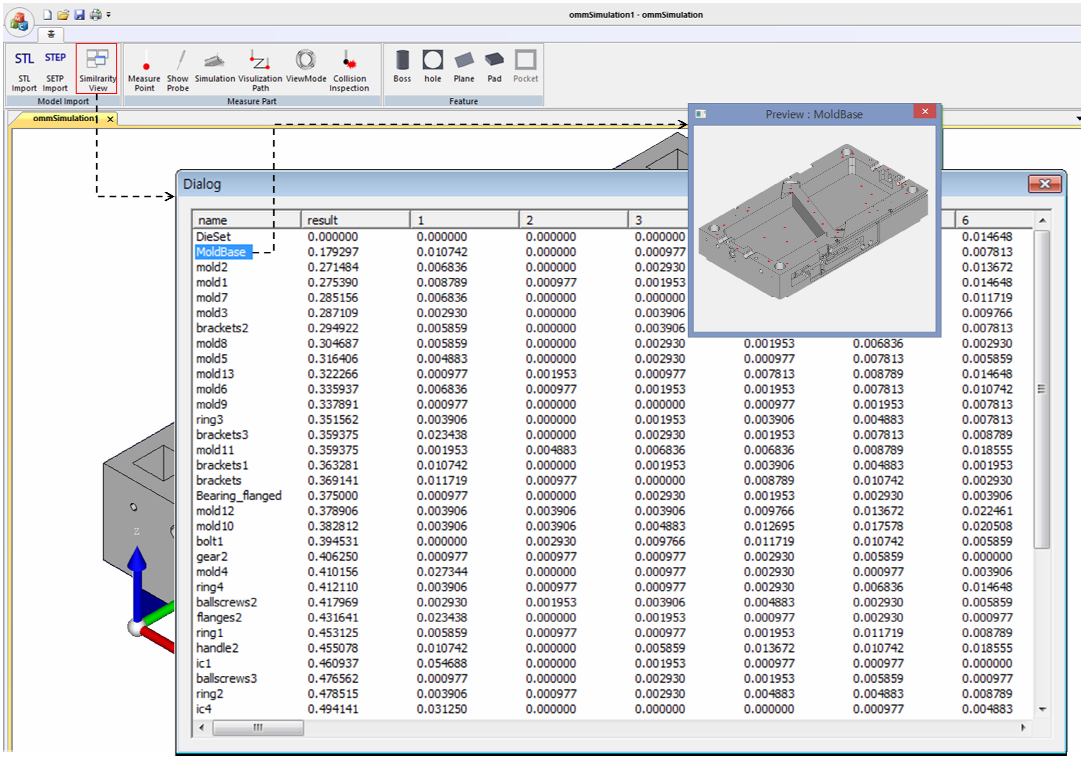



Fig. 13. Similarity analysis result of DieSet model.

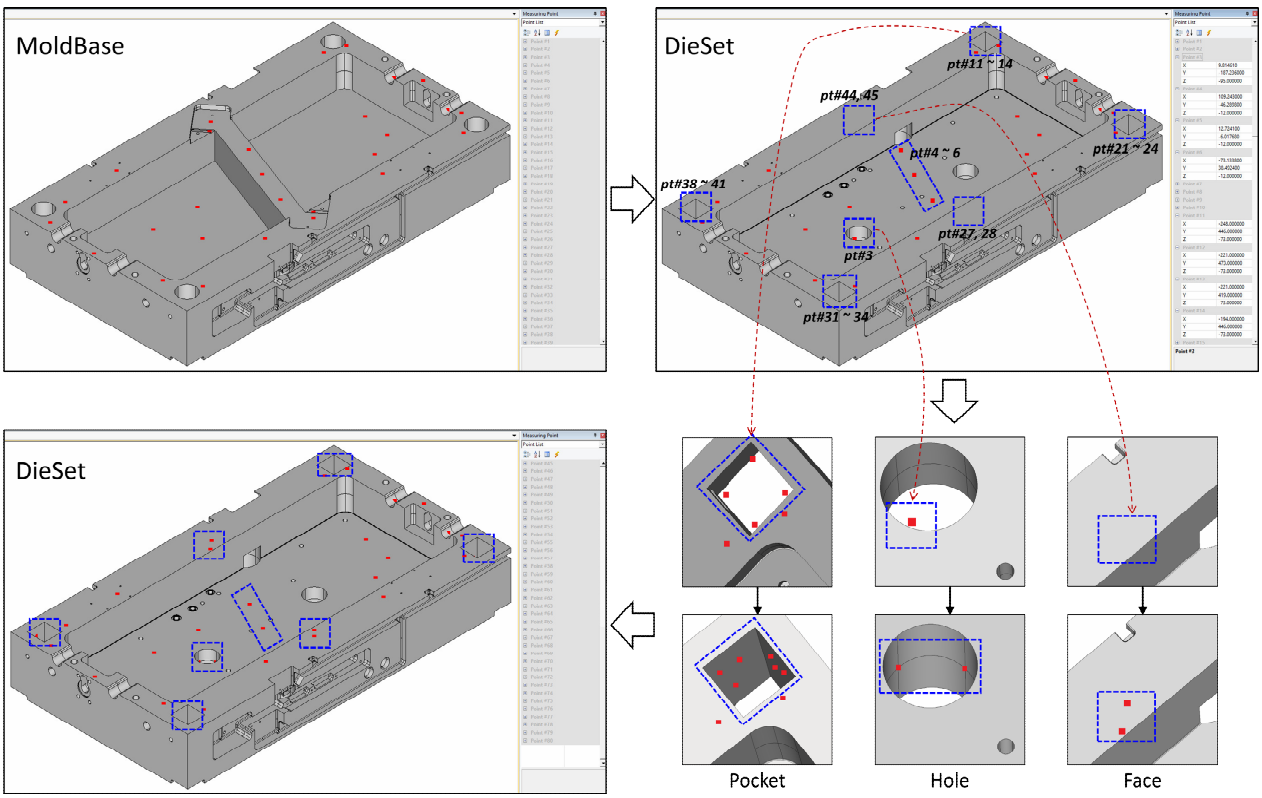



Fig. 14. Step-by-step application process of the measuring points of the MoldBase model to the DieSet model.

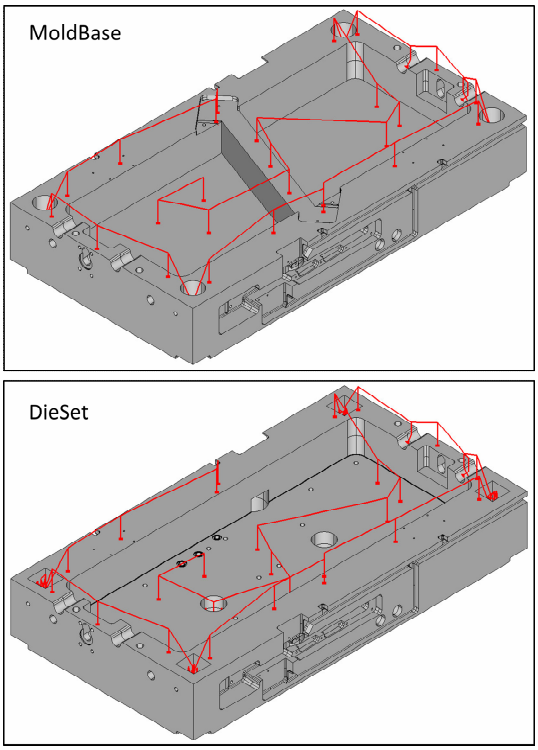



Fig. 15. Movement path of the MoldBase model (top) and of the DieSet model (bottom).

movement path, where the guide points and the offset points are set, it is required to modify the height of the guide points or relocate the offset points to the exterior of the model by performing a collision inspection simulation to examine the movement of the probe. Fig. 15 shows the result of the measurement path generation after the collision inspection is complete.

## 6. Conclusions

The OMM has the benefit of directly measuring a workpiece within the workspace by changing a tool into a probe on the machine without moving the workpiece during or after the machining process. However, it must create new measuring points and their path for every new model although it has similar feature shapes with existing models. Consequently, the measuring time increases and the measurement quality is affected by the skill of the engineers. To solve these problems, we suggest a touch-probe path generation method that employs similarity analysis between the feature vectors of a 3-D shape for the OMM. For the similarity analysis between a new 3-D model and existing 3-D models, we extracted the feature vectors from the models that can describe the characteristics of a geometric shape model and then applied them to a geometric histogram that displays a probability distribution according to the similarity analysis algorithm. In addition, we developed a CAIP system that corrects for non-applied measuring points caused by minute geometry differences between the two models and generates the final touch-probe path.

The proposed touch-probe path generation method has the advantage of improving the reusability of measuring points of existing models through similarity analysis. In this study, we included five feature shapes that are frequently used in the die/mold machining process. In future studies, we intend to explore a path generation method to easily apply various feature shapes, such as such as fillet, chamfer, and sweep, and their complex feature shapes along with the functional extension of our CAIP system.

## Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2015R1D1A1 A01060517).

## References

[1] S. Biasotti, S. Marini, M. Mortara, G. Patane, M. Spagnuolo and B. Falcidieno, 3D shape matching through topological structures, *Proc. of Discrete Geometry for Computer Imagery*, Naples, Italy (2003) 194-203.

[2] D. Bespalov, A. Shokoufandeh, W. C. Regli and W. Sun, Scale-space representation of 3D models and topological matching, *Proc. of the 8th ACM Symposium on Solid Modeling and Applications*, Seattle, USA (2003) 208-215.

[3] N. Iyer, Y. Kalyanaraman, K. Lou, S. Jayanti and K. Ramani, A reconfigurable 3D engineering shape search system part I: shape representation, *Proc. of ASME DETC 03 Computers and Information in Engineering Conference*, Chicago, USA (2003) 89-98.

[4] H. Sundar, D. Silver, N. Gagvani and S. Dickinson, Skeleton based shape matching and retrieval, *Proc. of Shape Modeling International*, Seoul, Korea (2003) 130-139.

[5] A. Elinson, D. S. Nau and W. C. William, Feature-based similarity assessment of solid models, *Proc. of the 4th ACM Symposium on Solid Modeling and Applications*, Atlanta, USA (1997) 297-310.

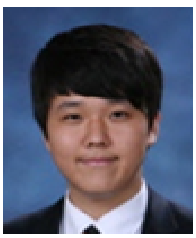
[6] D. McWherter, M. Peabody, A. C. Shokoufandeh and W. Regli, Database techniques for archival of solid models, *Proc. of the 6th ACM Symposium on Solid Modeling and Applications*, Ann Arbor, USA (2001) 78-87.

[7] A. Cardone, S. K. Gupta, A. Deshmukh and M. Karnik, Machining feature-based similarity assessment algorithms for prismatic machined parts, *Computer-Aided Design*, 38 (9) (2006) 954-972.

[8] S. Kim, K. Lee, T. Hong, M. Kim, M. Jung and Y. Song, An integrated approach to realize multi-resolution of B-Rep model, *Proc. of the 2005 ACM Symposium on Solid and Physical Modeling*, Boston, USA (2005) 153-162.

[9] R. Osada, T. Funkhouser, B. Chazelle and D. Dobkin, Shape distributions, *ACM Transactions on Graphics*, 21 (4) (2002)

807-832.
[10] C. Y. Ip, D. Lapadat, L. Sieger and W. C. Regli, Using shape distributions to compare solid models, *Proc. of 7th ACM Symposium on Solid Modeling and Applications*, Saarbrucken, Germany (2002) 807-832.
[11] W. Wohlkinger and M. Vincze, Ensemble of shape functions for 3D object classification, *Proc. of Robotics and Biomimetics*, Phuket, Thailand (2011) 2987-2992.
[12] R. Ohbuchi, T. Otagiri, M. Ibato and T. Takei, Shape-similarity search of three-dimensional models using parameterized statistics, *Proc. of Computer Graphics and Applications*, Beijing, China (2002) 265-274.



**Hyesung Jeon** has worked for Optimus Systems Inc. (www.optimus-sys.com), Korea. He received his master's degree in industrial system engineering in 2016 at Ajou University. His current research interests are geometric modeling, parametric feature modeling and human simulation.

**Jinwon Lee** is a Ph.D. student in the department of industrial engineering at Ajou University, Korea. He obtained his master's degree in industrial engineering in 2012 at Ajou University. His current research interests are geometric modeling, human simulation and 3-D simulation in virtual reality environment.

**Jeongsam Yang** is a Professor in the department of industrial engineering and is leading the CAD laboratory (http://cadlab.ajou.ac.kr) at Ajou University, Korea. He worked at Carnegie Mellon University (USA) and Clausthal University of Technology (Germany) as a visiting researcher, and the University of Wisconsin-Madison (USA) as a postdoctoral associate. He obtained his Ph.D. in mechanical engineering in 2004 at KAIST. His current research interests are product data quality (PDQ), VR application in product design, product data management (PDM), knowledge-based design system and geometry modeling.