Computer-Aided Design AND APPLICATIONS
www.cadanda.com

Kinetic Model Extraction from a Geometric Model

Sang C. Park[1] and Ji W. Oh[2]

[1]Ajou University, scpark@ajou.ac.kr
[2]Ajou University, ohjiwoong@ajou.ac.kr

ABSTRACT

Presented in the paper is a procedure to extract kinetic models from geometric models (manufacturing device models) consisting of multiple components and joints. Since the geometric model of a manufacturing device does not contain the kinetic mechanism, it is necessary to define the kinetic model for the virtual factory simulation. The function of the kinetic model is to operate joints. Because most manufacturing device models in the virtual factory consist of two joints including a prismatic joint and a revolute joint, this paper presents an algorithm for just two joints. A joint is created by contacting two components, and most of the contact surfaces consist of cylinder shapes. To detect contact surfaces of geometric parts, we use the concept of parallel computing using a GPU (Graphic Process Unit). Detected contact surfaces can be classified as a cylinder shape using the Gauss map. And the joints can be defined from the center axis of the cylinder shape. To classify joints, we also use the center axis and the collision detection. The proposed procedure has been implemented and tested with a simple fixture model in automotive body assembly lines.

Keywords: geometric model, kinetic model, gauss map, collision detection, contact detection.

1. INTRODUCTION

As technology develops, the computer simulation is used in a variety of fields including designing, executing and analyzing. Especially in the manufacturing industry, it is important that designing the product, process and resource in order to increase the competitiveness. These designs depend on the computer simulation. Many parts of manufacturing industry depend on the computer simulation, from a product design to production, as computing ability increases and people requires better virtual environments like realistic than ever. This demand has resulted in the concept of a virtual factory (VF). It is necessary to construct digital models for all the physical and logical elements (entities and activities) of a real manufacturing system to implement a virtual factory [3]. The fixture model and the 6-axis robot are typical digital models in the automotive body assembly lines, as shown in Fig. 1. If the digital model is not well implemented, we can't see the value in it. Accordingly, many algorithms have been developed about digital models.

The proposed algorithm in this paper uses many algorithms including the contact detection, the collision detection and gauss map. There are related researches. Tomas Moller proposed the algorithm for the intersection between triangles [5], and Min-Ho kyung did a research on the parallel computing of the GPU for the fast detection of the intersection between triangles [4]. Asada used the Jacobian matrix to model the fixture-work piece relationship in 3D space [1]. Samuel R. buss did a research about the Jacobian transpose method, the pseudo inverse method and the damped least squares methods to implement inverse kinematic [7]. Kang proposed the concept of the geometric model and the kinetic model [2]. In this paper we use this concept. Minsuk Ko proposed an algorithm for the extraction of the kinetic model using concept of 'moment of inertia' [3]. Weiwei Xu defines a joint of assembly model as well as considers deformation of shape of parts [8]. And Weiwei Xu uses many methods including the slippable motion analysis, eigenvalue. But this algorithm has some unnecessary processes to apply CAD assembly model in virtual factory.

To perform various works, all of the digital models have different shapes. But most of them have just two joints including revolute joint and prismatic joint. Because of this, we ignore every joint except mentioned two joints. Commercial soft wares like 'CATIA' require much effort for defining kinetic model because user chooses the contact surface of each

Taylor & Francis
Taylor & Francis Group

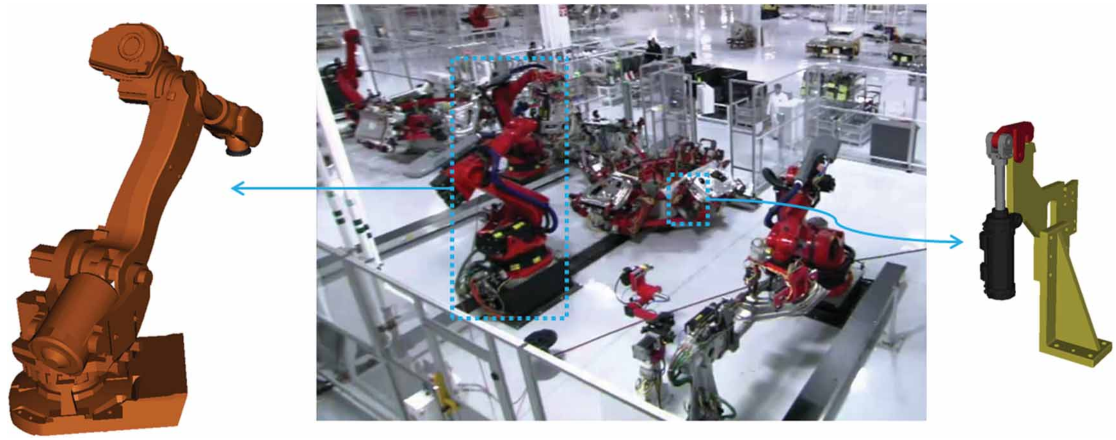

Fig. 1: Models that require kinetic model in virtual factory.

geometric component. As product design changes frequently in the automotive body, re-defining the kinetic model is a waste of time and money. In this paper, the proposed algorithm can eliminate the waste of time and money that be created when user re-defines the kinetic model.

The remainder of this paper is organized as follows. Section 2 presents the overall approach to the automatic creation of a kinetic model from a geometric model. Section 3 details the proposed algorithm. Section 4 presents concluding remarks.

2. APPROACH TO THE AUTOMATIC CREATION OF A KINETIC MODEL

Figure 2 shows the assembly geometric model that requires kinetic model in virtual factory. As the aforementioned assembly geometric model consists of rigid geometric parts, the joint of the assembly geometric model is made by contact of each rigid geometric part. To perform various works, all of geometric model have different shapes. Even though the shape of the geometric model is very complex, most of the assembly geometric models have just two joints (revolute joint, prismatic joint). Many activities including assembly, welding, fixing and examination are performed with these two joints in the virtual factory.

To do so, it is necessary to go through three important steps: (1) contact detection; (2) identification of the contact surface of cylinder shape and creation of the center axis; (3) defining of the joint using the collision detection.

For the first step, OBB (Oriented Bounding Box) binary tree is applied to the contact detection, and we use parallel computation of the GPU for a fast

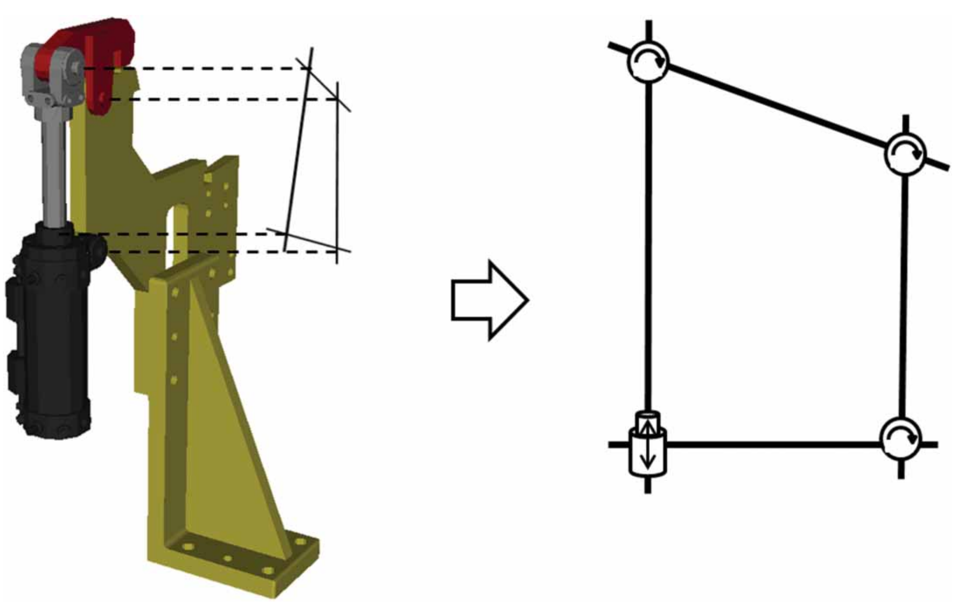

Fig. 2: Kinetic model extraction from a geometric model.

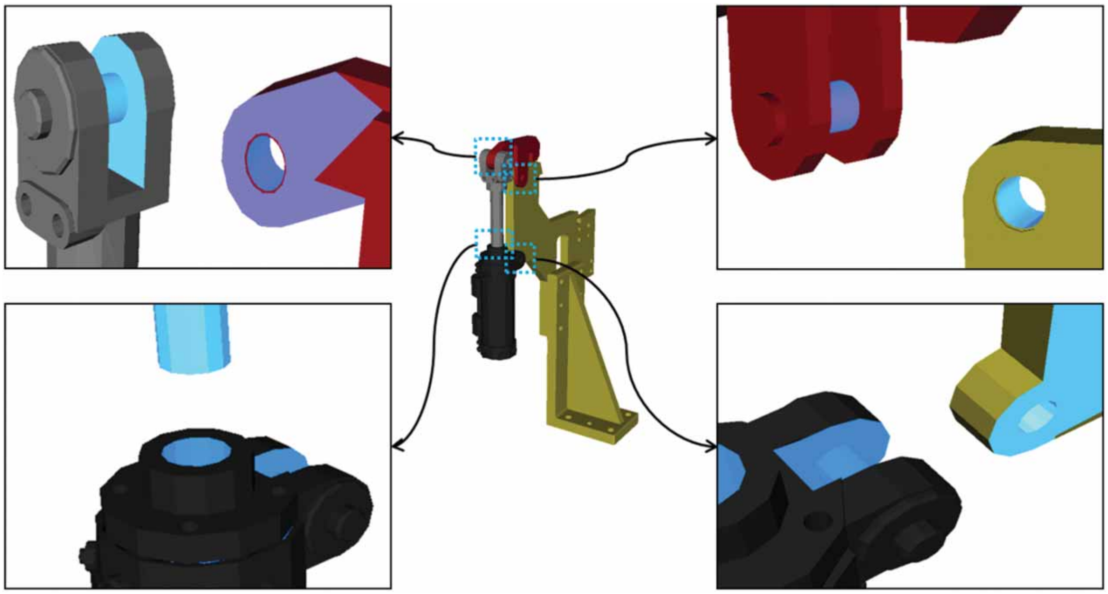

Fig. 3: Contact detection from geometric model.

computing. OBB binary tree is a structure for the quick calculation through the collision detection of boxes that can cover each geometric part. In addition, parallel computation can be used when many calculations are required. When we detect contact surfaces of geometric model by using parallel computing, many triangles can be calculated at the same time. As a result, parallel computing can improve speed of calculation. The detected contact surface is shown in Fig. 3, and we can see the contact surface of the cylinder shape at the joint.

The second step is the identification of the contact surface of the cylinder shape and creation of the center axis. This is the reason why most of mentioned two joints (revolute, prismatic) has the contact surface of the cylinder shape that we have to identify. Therefore, it is necessary that you identify the cylinder shape by using the Gauss map in order to define the joint. The next section addresses a detailed explanation of the Gauss map. After identifying the cylinder shape, you can see the center axis of the cylinder shape created by a center point and a normal vector of triangle that forms the contact surface as shown in Fig. 4. This center axis is necessary for deciding type of joint and defining a motion of kinetic model.

We decide type of joint by using collision detection in the third step. The generated center axis is used to decide the type of joint. A collision between two parts that consist of revolute joint occurs when one of the parts moves in the direction of the center axis or in the reverse direction of the center axis. Because revolute joints provide single-axis rotation function and prismatic joints provide single-axis linear sliding movement function. This method is algorithm proposed in this paper and the next section addresses a detailed explanation of the algorithm by using collision detection.

3. ALGORITHM TO EXTRACT A KINETIC MODEL

The proposed procedure consists of two steps; (1) detection of the contact surfaces of the cylinder shape and an algorithm to generate center axis; and (2) an algorithm to decide the type of joint.

In the first step, we've used the gauss map to generate the center axis and to detect the contact shape of the cylinder shape. This method shows the point on the unit sphere. The point comes from the normal vector on the surface consisting of a triangular net. If the circle shape is illustrated on the unit sphere when we apply a normal vector of contacted triangles to the gauss map, the contact surface of cylinder shape can be defined. The unit normal vector illustrates a point on the unit sphere only in the contact flat surface. And we have to do cross product between normal vectors in order to generate the center axis. Because when we perform the cross product of the normal vectors of the cylinder, we create the new center axis as shown in Fig. 4-(b)

This vector is necessary to identify the type of joints and to define the motion of joints. Fig. 5 shows the center axis which is generated by detecting the contact surface of the cylinder shape in the fixture model.

The second step is algorithm to define the type of joints with collision detection. This paper focuses on aforementioned two joints including a revolute joint and a prismatic joint. The revolute joint provides not linear sliding movement function but single axis rotation function. On the other hand, the prismatic joint

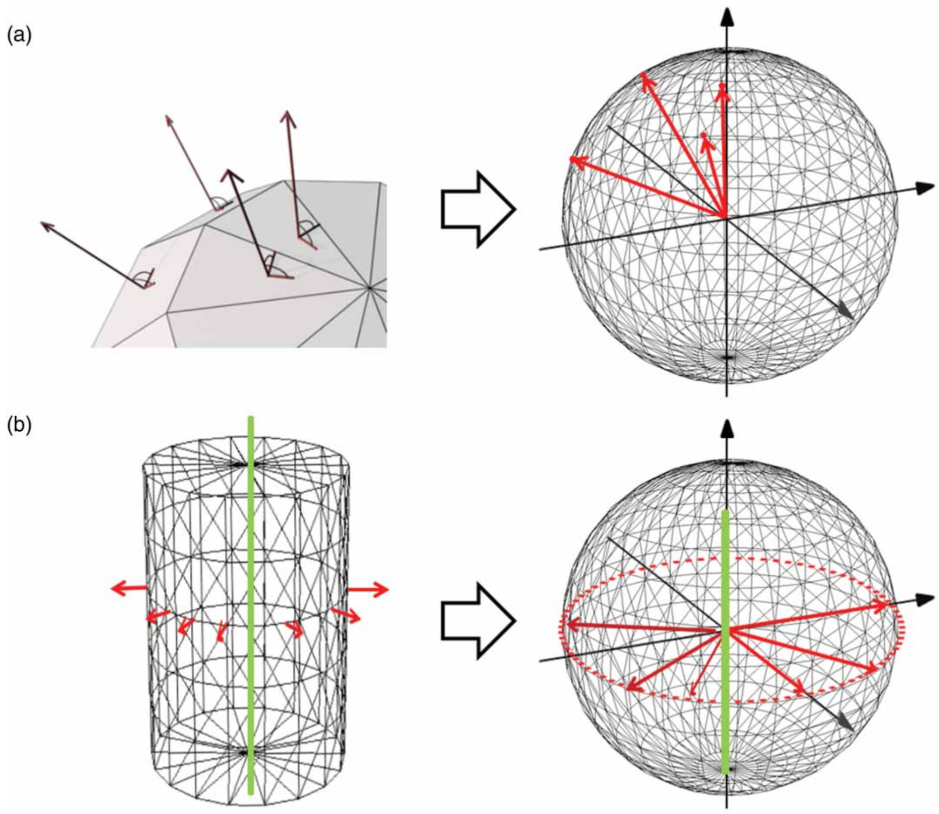

Fig. 4: Cylinder shape detection using Gauss map.

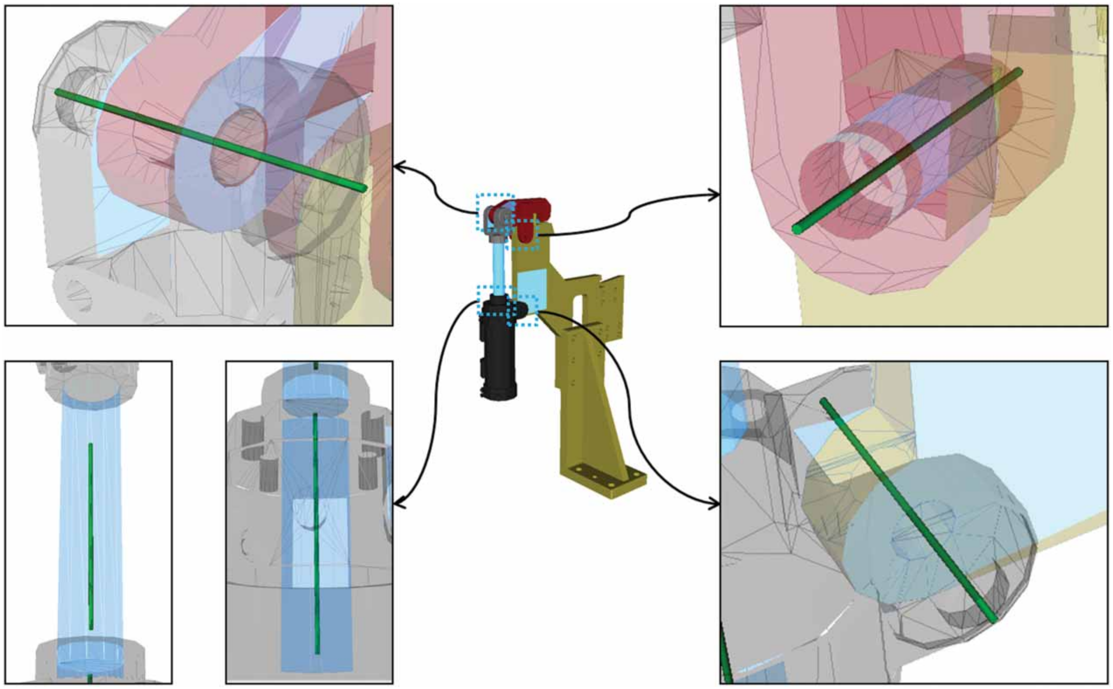

Fig. 5: Center axis creation from contact detection.

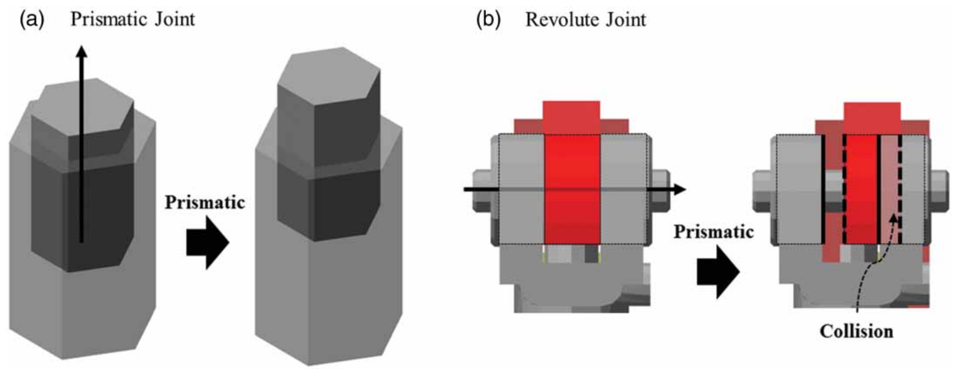

Fig. 6: Joint classification using collision detection.

provides single-axis linear sliding movement function. Fig. 6 shows that a characteristic of this joint is that they do not collide when they do a rotary motion. If there is a detect collision when we move one of parts of the joint into the center axis vector direction and reverse direction, this joint is a revolute joint. We can know whether it is collision through the distances between parts after moving. If the change of the distance between parts, this is collision.

The problem of defining the type of joints can be described as follows:

- ***NP***. A number of the geometric parts of assembly model.
- ***P.*** A one of parts, if $1 \le i \le NP$, a ith part is $\boldsymbol{P_i}$
- ***Vec(P).*** A unit vector of *P*, if *P* have the contact surface of the cylinder shape.
- ***PCP.*** A container that have pairs of contacted parts. if $1 \le i \le$ *PCP.num,* a *i*th *PCP* is *PCP[i]*
- ***PCP.num. A*** number of *PCP*.
- ***PCP.insert(P_i, P_j).*** Instruction that put the pair of contacted parts***(P_i, P_j)*** into the *PCP*.
- ***PCP.joint.*** Definition of the joint.
- ***CollisionDetection(PCP).*** Judgment of whether two parts in *PCP* are collision when parts moving into direction of *Vec(P)*and reverse direction.

```
//Cylinder shape contact detection
For (i = 1; i < NP; i ++)
{
  For (j = 1; j ≤ NP; j ++)
  {
    If (P_i and P_j do not contact)
      Continue;
    If (P_i and P_j do not cylinder shape contact using
    gauss map)
      Continue;
    If(Vec(P_i) = = Vec(P_j))
      PCP.insert(P_i, P_j);
  }
}
//Definition of joint
For (i = 1; i ≤ PCP.num; i ++)
{
  If(CollisionDetection(PCP[i]_1))
    PCP[i].joint = Revolute joint;
  Else
    PCP[i].joint = Prismatic joint;
}
```

It is possible to extract the kinetic model from assembly geometric model using the proposed algorithm and to implement the motion of model using the center axis.

4. SUMMARY

Manufacture industry requires automated robots assuming variety roles. These robots consisting of geometric parts perform many roles including welding, assembly and inspection using their joints. Most of the joint type of manufacturing devices is either one of the revolute or prismatic types.

There are various virtual devices in a virtual factory. Each virtual device consists of two sub-models; a geometric model and a kinetic model. A geometric model is usually provided by a vendor company in the form of a visible CAD model.

To define the kinetic model, we have to select the contact surface, axis of the geometric model and type of joint. If the virtual device has complex shapes, time and effort are needed to define the kinetic model. Because it is inevitable that the product design frequently changes in the manufacture industry, defining the kinetic model is a waste of time and money. This paper proposes an algorithm for the automatic extraction of the kinetic model from the assembly geometric model to reduce waste. This algorithm has important 3 steps: (1) contact detection; and (2) identification of the contact surface of the cylinder shape and generation of the center axis of joints; (3) definition of joints using collision detection. Both the OBB binary tree structure and parallel computing of the GPU increased the speed of detection of the contact surface, and we used the gauss map in order to identify the cylinder shape and generate the center axis. Most of joints including prismatic joint and revolute joint can operate because joints have the contact surface of the cylinder shape. And joints are defined using collision detection.

Although this algorithm is applied only to the fixture model in this paper, it can be applied to the various categories. We will consider automatic extraction of the kinetic model from not only two joints (revolute, prismatic), but also the various joints.

REFERENCES

[1] Asada, H.; By, A.: Kinematic analysis of workpart fixturing for flexible assembly with automatically reconfigurable fixtures, IEEE Journal of Robotics and Automation, 1(2), 1985, 86–94. http://dx.doi.org/10.1109/JRA.1985.1087007

[2] Kang, Y.; Rong, Y.; Yang, J. A.: Geometric and kinetic model based computer-aided fixture design verification, Journal of Computing and Information Science in Engineering, 3(3), 2003, 187–199. http://dx.doi.org/10.1115/1.1607352

[3] Ko, M. S.; Park, S. C.: Fixture modeling for an automotive assembly line, International Journal of Production Research, 49(15), 2011, 4593–4604. http://dx.doi.org/10.1080/00207543.2010.506893

[4] Kyung, M. H.: Robust GPU-based intersection algorithm for a large triangle set, Conference of Korean Computer Graphics, 17(3), 2011, 9–19. http://dx.doi.org/10.15701/kcgs.2011.17.3.9

[5] Moller T.: A fast triangle-triangle intersection test, Journal of Graphics Tools, 2(2), 1997, 25–30. http://dx.doi.org/10.1080/10867651.1997.10487472

[6] Park, S. C.: A methodology for creating a virtual model for a flexible manufacturing system, Computers in Industry, 56(7), 2005, 734–746. http://dx.doi.org/10.1016/j.compind.2005.04.002

[7] Samuel R. B.: Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods, 2009, http://math.ucsd.edu/~sbuss/ResearchWeb.

[8] Xu, W.; Wang, J.; Yin, K.; Zhou, K.; Panne, M.; Chen, F.; Guo, B.: Joint-aware manipulation of deformable models, SIGGRAPH ’09, 28(35), 2009, 1–9.