Talon Metaphor: Grasp and Release Method for Virtual Reality

Yusik Kim[1] and Jung-Min Park[2]([✉])

[1] Korea Advanced Institute of Science and Technology, Daejeon, Korea
`usik.02@gmail.com`
[2] Korea Institute of Science and Technology, Seoul, Korea
`pjm@kist.re.kr`

Abstract. In this paper, we present a novel grasp and release method for direct manipulation in virtual reality. We develop a new grasp method in complementary to the proposed release method. Data gloves and smart sensors may achieve high accuracy and speed but users are prone to fatigue after a long use and devices can be expensive. Therefore, we focus our study to develop an algorithm to reduce fatigue and costs of the virtual system. We track the real hands using a depth camera and determine grasp and release states using virtual rays. Without wearing any devices, users are able to grasp and release a virtual object quickly and precisely. Just as eagles use their talons to catch prey, finite virtual rays from the finger tips are used to determine the grasp and release states. We present that our method improves on grasp time, release time and release translational error based on grasp and release tasks. In contrast to the natural interaction metaphor, measured grasp time, release time and the release translational error for our method do not depend on the object types and sizes.

Keywords: Virtual reality · Virtual grasp · Bare-hand interaction

1 Introduction

In recent years with the advancement of depth cameras, mid-air, bare-hand or free-hand interaction has received significant attention from the research community. Commercially available devices such as Kinect and WiiMote allowed researchers to produce interaction methods in virtual reality. Generally, there are two ways to provide the input for mid-air interaction. The first one is by wearing data gloves, sensors or markers that provide hand information to the system. The other one is device-independent bare-hand method, which does not require any wearable devices. Depth camera obtains the hand information and determines different poses of the hand in real time. One example of mid-air interaction is the natural interaction metaphor [1], in which users wear finger tracking gloves in a virtual reality system. Another example is the spring model, in which users wear data gloves for mid-air interaction in virtual environment [2–4]. Although previous works provided good solutions for manipulating an object in virtual environment, sticking object problem mentioned in [1] and tracked fingers residing inside of an object problem in [2–4] hinder users from experiencing naturalness, which is one of the requirements to satisfy from user's perspective [5]. According to

Prachyabrued and Borst [3], both problems occur due to lack of physical constraints that users tend to close their real hands into the virtual objects. One way of solving the prob-lems is by wearing devices and sensors that give feedback to users when they touch the object. Many techniques take advantage of wearable data gloves, sensors and markers to track hands accurately and quickly but the wearable devices, sensors and markers can hinder users from being focused and experiencing naturalness. Our method only uses one RGBD sensor for both tracking hand and manipulating a virtual object yet, it enhances naturalness and reduces fatigue on users by solving sticking object problem and residing inside of an object problem. Similarly gesture based techniques, such as the handle bar metaphor [6], only require one RGBD sensor to manipulate multiple objects at once but users need to learn the predefined gestures before they can take advantage of such virtual systems. Our algorithm calculates optimal grasping and releasing states using finite virtual rays, which simply calcula-tions and provide rapid and accurate selections in comparison to natural interaction metaphor. As long as users are familiar with grasp and release in real world, this method has low learning curve for average people to enjoy virtual reality tasks. In this paper, we present talon metaphor for bare-hand virtual grasp and release, which does not rely on any wearable devices, sensors and markers. We present experimental evaluation of our method to show speed and accuracy in comparison to natural interaction metaphor.

2   Talon Metaphor

2.1   Talon Metaphor Description

When we were researching about grasp and release mechanisms in virtual reality, we decided to take careful examination at biological beings in real world. We named our metaphor as talon after the claws on birds' feet because eagles, specifically, can grab and lift up their prey from water effortlessly. We were impressed at how quickly and firmly eagles can catch their prey. Therefore, we implemented finite virtual rays on virtual fingertips that mimics talons on eagles' feet in Fig. 1. Finite virtual rays project out from the middle of the tips of thumb, index and middle finger as shown in Fig. 3. When eagles grasp their prey, their talons penetrate the body of the prey. Similarly, our method calculates the intersection points of two pairs of virtual rays inside of an object for grasp. As eagles can rapidly and accurately grasp and release their prey, our metaphor also can grasp and release quickly and accurately.

2.2   Grasp Function

Our grasp function renders virtual rays from the middle of the three finger tips per-pendicular to their distal joints. The basic idea is to check the shortest distance between the rays and compare it with the predefined threshold. The steps are shown in Algo-rithm 1. Figure 2 illustrates the moment of grasping an object. Let:

Fig. 1. Talon clenching fist.

$H \leftarrow$ User's Hand Coordinates,
$O \leftarrow$ Object's Coordinates,
$j \leftarrow$ longest length in the bounding box,
$k \leftarrow 2^n d$ longest length in the bounding box,
$d_{tm} \leftarrow$ distance between thumb and middle tip,
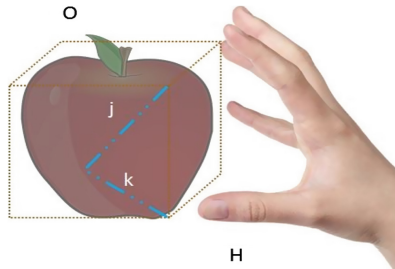*palmSize* $\leftarrow$ average palm size.

Fig. 2. Virtual rays are calculated and rendered from the midpoint of finger tips perpendicular to distal joints.

Algorithm 1 Grasp–Release algorithm
1: function Grasp–Release
2:   while H contacts O do
3:     CreateBoundingBox
4:     CalculateRayLength
5:     RenderRay
6:     CheckGraspingPairs
7:   end while
8: end function

Step 1: While H contacts O, create a bounding box. Our method checks for the moment when the user's hand contacts an object. Contact is valid when the user's thumb and one of the fingers are touching the object. Since we know the co-ordinates of the object and the hand, we can simply determine whether the contact is valid or not. When the contact is valid, a bounding box, which surrounds the object, is calculated using 3D point cloud coordinates of the object.

Step 2: Calculate length of the ray. This step as shown in Algorithm 2 calculates and renders the rays that are projected from the middle of the three finger tips. First we calculate the distance between the tips of thumb and middle finger, d_{tm}. If d_{tm} is larger than *palmSize*, it is considered to be wide grasp. Otherwise, it is considered to be narrow grasp. If it is the wide grasp, we set the ray length to be the half of the maximum length of the bounding box, which is j, otherwise we set the ray length to be the half of k. Rays are projected away from the palm at 90° from the distal joints. Figure 3 shows how rays are rendered.

Step 3: Check for grasping pairs. This step checks for valid grasping pairs and returns true if the grasping conditions are satisfied. First grasping condition is checking for a valid grasping pair. A valid grasping pair is either a pair of thumb and index finger or a pair of thumb and middle finger. Then the shortest distance between the virtual rays of valid grasping pairs is calculated and compared with the predefined threshold value. If the shortest distance is equal to or less than the threshold value, the second grasp condition is satisfied. Finally, an object is grasped when the two grasping conditions are satisfied. At each simulation frame that user is grasping an object, our system cycles through the above steps checking for validity of grasping conditions. Step 3 is shown in Algorithm 3.

Algorithm 2 CalculateRay Length

1: function CalculateRayLength
2:   if check WideGrasp( ) is true then
3:     $L_k \leftarrow j/2$
4:   else
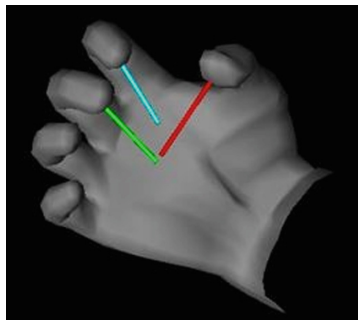5: $L_k \leftarrow j/2$
6:   end if
7: end function

Fig. 3. Virtual rays are calculated and rendered from the midpoint of finger tips perpendicular to distal joints

Algorithm 3 CheckGraspingPairs
1: function CheckGraspingPairs(R_{thumb}, R_{index}, R_{middle})
2: if getShortestDistance(R_{thumb}, R_{index}, R_{middle}) $\leq$ Threshold then
3:    ShortCoords $\leftarrow$ getShortestDistanceCoords
4:    if shortCoords in Object then
5:      Grasp
6:    end if
7: else
8:    Release
9: end if
10: end function

2.3 Release Function

Grasping of an object is valid as long as there is at least one valid grasping pair associated with the object. In each simulation frame, our system compares the predefined threshold with the shortest distance between the grasping pair as shown in Algorithm 3. If the check for grasping pairs returns false during grasp, the release state begins. However, we noticed unwanted drops of objects occurred in our system due to the finger sensing noise and unintentional movements. In order to solve the problem, we implemented low-pass filter. Note that the low pass filter level can be changed respect to the hardware.

3 Experiment

3.1 Experimental Setup

As shown in Fig. 4, our system consists of a desktop with Intel I7-4790 QuadCore at 3.6 GHz and 32 GB memory running on Windows 8 with NVidia GeForce GTX 780 graphics board. A Prime-Sense camera is mounted on the top facing downward to tack user's hands. A 40-inch 3D flat panel display is used as the monitor for 3D functionality with the compatible 3D glasses.
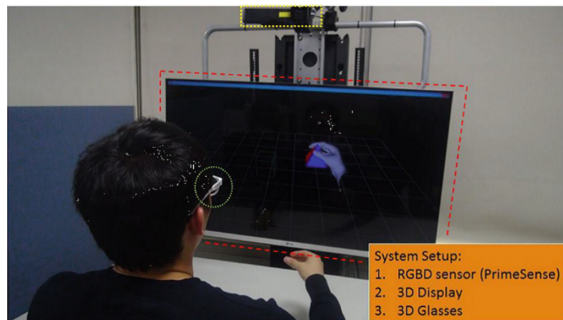

Fig. 4. System setup

3.2   Experimental Results

We measure the time to grasp an object, the time to release an object and translation error of an object during release using the talon metaphor. The measurements were compared with those of the natural interaction metaphor, which we implemented by following the descriptions on published work [1]. Natural interaction metaphor defines a collision pair which is a pair of two fingers holding an object. Collision pairs define a valid grasping pair when the friction angle of both collision pairs is smaller than a predefined threshold. When a user is grasping an object, the natural interaction metaphor calculates the distance between the grasping pair and the midpoint of the grasping pair, which is called the barycenter. The first release condition is when the distance between the grasping pair is larger than the predefined threshold. The second release condition is when the distance between the barycenter and the object's center is larger than the allowed threshold. If one of the two release condition gets violated, the release begins. We experiment with grasp and release task on various object size (small (length or diameter = 4.0 cm), medium (length or diameter = 7.0 cm), large (length or diameter = 10.0 cm)) and various object types (cube, polyhedron, sphere).

We performed an experiment with 16 males and 4 females, aged 23 to 31 years (average = 27). Almost all subjects graduated from engineering majors and had exposure to virtual reality systems.

We compared the results of our method to the results of the natural interaction metaphor and summarized in Table 1. Regard-less of the object sizes and types, our method was on average 30% faster than the natural interaction metaphor at grasping an object, on average 70% faster at releasing an object and on average 18% less in release translational error.

Table 1. Grasp time, release translation error, release time for different object sizes and types compare to natural interaction metaphor (unit = ms)

	Method	Size			Type		
		Small	Medium	Large	Cube	Polyhedron	Sphere
Grasp time	Natural interaction	1843.46	1593.02	1990.34	1764.77	2145.72	1516.33
	Talon	1221.96	1233.38	1284.21	1146.43	1292.25	1300.87
Release translation error	Natural interaction	18.13	23.30	25.61	22.46	26.15	19.13
	Talon	17.44	17.58	18.77	18.93	17.53	17.33
Release time	Natural interaction	1512.22	1637.63	2155.36	1636.96	2029.57	1638.68
	Talon	441.54	500.45	456.07	462.64	448.06	489.36

4   Conclusion

We propose a new grasp and release method called talon metaphor in virtual reality. We only use a commercially available and affordable RGBD sensor to track the real hands. It can directly manipulate an object without wearing any devices in virtual reality. The experimental results show that the talon metaphor successfully solves sticking object problem and fingers residing inside of an object problem and significantly improves in grasp and release time as well as release translational error. Using our method, normal users can enjoy and explore manipulation tasks that require precision and quick grasp and release method in virtual reality.

References

1. Moehring, M., Froehlich, B.: Natural Interaction Metaphors for Functional Validations of Virtual Car Models. IEEE Trans. Vis. Comput. Graph. **17**(9), 1195–1208 (2011)
2. Prachyabrued, M., Borst, C.W.: Dropping the ball: releasing a virtual grasp. In: IEEE Symposium on 3D User Interfaces (3DUI 2011), pp. 59–66, Singapore (2011)
3. Prachyabrued, M., Borst, C.W.: Virtual grasp release method and evaluation. Int. J. Hum.-Comput. Stud. **70**, 828–848 (2012)
4. Prachyabrued, M., Borst, C.: Visual feedback for virtual grasping. In: IEEE Symposium on 3D User Interfaces (3DUI 2014), Minneapolis, Minnesota, USA, pp. 19–26 (2014)
5. Jankowski, J., Hachet, M.: A survey of interaction techniques for interactive 3D environments. In: Eurographics 2013 (EG 2013), pp. 65–93 (2013)
6. Song, P., et al.: A handle bar metaphor for virtual object manipulation with mid-air interaction. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2012), Austin, Texas, pp. 1297–1306 (2012)