

Article

# Singulation of Objects in Cluttered Environment Using Dynamic Estimation of Physical Properties

Abid Imran <sup>1</sup>, Sang-Hwa Kim <sup>2</sup>, Young-Bin Park <sup>3</sup>, Il Hong Suh <sup>3</sup> and Byung-Ju Yi <sup>1,\*</sup><sup>1</sup> Department of Electronic Systems Engineering, Hanyang University, Ansan 15588, Korea<sup>2</sup> Department of The Division of Electrical Engineering, Hanyang University, Ansan 15588, Korea<sup>3</sup> Department of Electronics and Computer Engineering, Hanyang University, Seoul 04763, Korea

\* Correspondence: bj@hanyang.ac.kr

Received: 12 July 2019; Accepted: 23 August 2019; Published: 28 August 2019



**Featured Application:** Singulation of objects in a cluttered living room environment to enhance the grasping and identification capabilities of a robot manipulator. Singulation of objects in autonomous assembly by robot manipulator.

**Abstract:** This paper presents a scattering-based technique for object singulation in a cluttered environment. An analytical model-based control scattering approach is necessary for controlled object singulation. Controlled scattering implies achieving the desired distances between objects after collision. However, current analytical approaches are limited due to insufficient information of the physical environment properties, such as the coefficient of restitution, coefficient of friction, and masses of objects. In this paper, this limitation is overcome by introducing a technique to learn these parameters from unlabeled videos. For the analytical model, an impulse-based approach is used. A virtual world simulator is designed based on a dynamic model and the estimated physical properties of all objects in the environment. Experiments are performed in a virtual world until the targeted scattering pattern is achieved. The targeted scattering pattern implies that all objects are singulated. Finally, the desired input from the virtual world is fed to the robot manipulator to perform real-world scattering.

**Keywords:** object singulation; clutter environment; dynamic estimation; impact mechanics; physics engine; dynamic manipulation

## 1. Introduction

Robot grasping is well enough developed for a structured environment [1,2]. However, interaction in unstructured environments is quite a tedious task for robots. Different techniques have been proposed for object manipulation in cluttered environments, such as data driven approaches [3] and using a hierarchy of supervisor for online learning from demonstration [4,5].

In robot manipulation, the objects can be moved from initial position to goal position through quasi-static or dynamic manipulation. The object singulation can be achieved by either pushing (quasi-static manipulation method) or scattering (dynamic manipulation method) the objects. In this paper, the methodologies are developed for object singulation, considering the dynamic manipulations. Object singulation assists grasping objects in a cluttered environment by creating fewer obstacles and more clearance for the robot end effector. Especially, when objects are close to each other or come into contact, it is hard for the robot to grasp an individual object without separating it from other objects. That is why it is sometimes necessary to scatter objects by hitting them with a manipulator for a short period of time to facilitate a grasping operation. Such cluttered environments can be found in our living rooms, as shown in Figure 1a. The task will be identifying objects, grasping, and finally

rearranging them at their original locations. The singulation of objects will help in identification and grasping of objects. Another emerging issue is the autonomous assembly by a robot manipulator. Lego parts based assembly example is shown in Figure 1b. Many objects are mixed together, and the robot performs the object singulation and then should identify different parts in sequence in real-time, grasp it, and complete the assembly.



**Figure 1.** Potential applications of singulation: (a) Living room; (b) LEGO parts-based assembly by a robot manipulator.

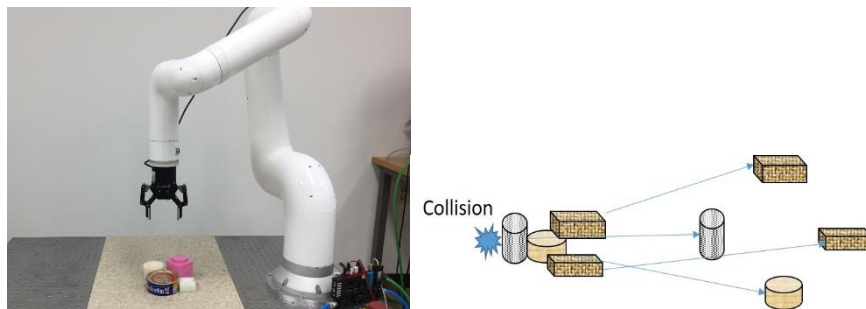
Different methodologies of object singulation are employed by using a pushing motion in cluttered environment [6–10]. Eitel et al. [7] proposed a neural network-based approach to separate unknown objects in clutter through favorable push actions. By performing selective pushes, Hermans et al. [8] proposed a method for separating an unknown number of objects by disambiguating potential object boundaries. Dogar et al. [11] used a push-grasping technique to enhance the grasping capabilities of the robot by introducing the concept of capture region.

Standard model-based approaches are also applied for object singulation [12–14] in a cluttered environment. The model-based approaches require knowledge of physical properties of the environment. Without any prior knowledge, the estimation of the physical properties is a challenging task. However, the physical properties can be estimated by processing the recorded video of dynamic interaction of objects along with the analytical model. Wu et al. [15,16] proposed a technique to acquire physical properties of objects from unlabeled videos. However, this technique is not directly applicable for an object singulation application.

Impulse model-based methodologies are introduced for the robot and environment interaction applications [17–19]. By extending those methodologies, we propose an analytical model based scattering technique for object singulation in a controlled manner. Controlled scattering implies achieving a desired distance among objects after collision as shown in Figure 2. The robot performs scattering by hitting the object nearest to the gripper. As a result, the nearest object collides with the surrounding objects and makes it easy for the robot to pick up the target object. In this paper, an impulse-based model is used for analytical modeling of the environment. A robot manipulator will be used as a means of impulse given to the object. The problem of physical properties estimation is overcome by using a vision-based technique. A virtual world simulator is developed to perform a controlled scattering without disturbing the real-world environment.

In Section 3, a generalized impulse model is introduced. Section 4 describes a methodology to acquire motion parameters such as distance, time, velocity, and accelerations from unlabeled recorded videos. These parameters are used in a dynamic model to learn the physical properties such as mass ratio, coefficient of restitution, coefficient of friction. In Section 5, a virtual world simulator is designed to perform the controlled scattering. Once the controlled scattering is achieved in virtual world, the manipulator will be commanded to perform the scattering similar to virtual world. Finally, a quantitative evaluation is performed in Section 6 by measuring the distance among

objects in the real and virtual world to evaluate the performance of the proposed scattering method and exactness of the estimated physical properties.



**Figure 2.** Concept of scattering based object singulation.

## 2. Methodology

In robot manipulation, there are two different methods to move objects from initial position to final position, such as prehensile and non-prehensile methods. In prehensile manipulation, the method is to grasp an object and move to final position. On the other hand, in non-prehensile manipulation, the object can be moved to goal position by pushing, throwing, or striking. Furthermore, non-prehensile manipulation methods are mainly divided into two categories, such as quasi-static and dynamic methods [20]. In quasi-static manipulation, an object is always in contact with a manipulator and motions are assumed slow enough to neglect the inertial effect, such as pushing [21] and sliding [22]. The quasi-static manipulation is not suitable for limited workspace and in some applications where manipulations of an object demand higher speed. On contrary, in the dynamic manipulation, the object continues its motion after losing contact with manipulator. In the dynamic manipulation, the motion of the object is determined by the motion of manipulator and its dynamic characteristics after losing contact. This paper deals with singulation of objects in cluttered environment based on dynamic manipulations, where impulsive motions are used to scatter the object to achieve the desired distance among objects after scattering. Practically, it is difficult to control the moving object with impulsive actions to an exact location however, it is still useful for object singulation application as the targets are not that stringent.

In this paper, 115 videos were collected by performing the collision of single object and multiple objects with a manipulator. By processing these unlabeled videos, certain parameters are observed such as travel distance of each object after collision, travel time, velocities before/after collision, and acceleration of each object. Furthermore, using these observed motion parameters, physical properties of all objects being involved in the collision can be estimated. Based on these physical properties, a virtual world simulator is designed to perform a controlled scattering. In control scattering, the maximum and minimum distance is controlled between objects after collision to ensure that all objects are singulated. The initial position of objects is observed from the real world and is given as an input to the virtual world along with measured physical properties. The distance between objects is a function of manipulator colliding velocity. In the virtual world, the velocity of manipulator is controlled based on the feedback error (between measured distance and desired distance) until the desired distance among objects is achieved. Once the goal is achieved in virtual world, the same desired velocity input is given to the real world and then controlled scattering is performed.

## 3. Impulse Modeling

In this section, the analytical impulse based methodology is developed considering the dynamic manipulation of objects. During scattering, the impulse is propagated from robot manipulator to the objects involved in scattering. Accordingly, it is important to develop a model considering the collision between the robot manipulator and object, and between objects.

### 3.1. Collision between Manipulator and Object

The scattering environment is shown in Figure 3. The incremental change in relative linear velocities of two colliding bodies (here, a robot and a colliding body) can be calculated if the coefficient of restitution ( $e$ ) is known [23]:

$$(\Delta v_I - \Delta v_{env})^T \mathbf{n} = -(1 + e)(v_I - v_{env})^T \mathbf{n}, \tag{1}$$

where the coefficient of restitution ( $e$ ) ranges from 0 to 1 correspond to perfectly inelastic to elastic collision.  $v_I$  and  $v_{env}$  are the absolute linear velocities of the robot and colliding body (environment), respectively.  $\Delta v_I$  and  $\Delta v_{env}$  are the velocity increments of robot manipulator and the colliding body after the impact, respectively. The inverse dynamic model of the robot manipulator with respect to independent joint set is given as follows [24]:

$$\mathbf{T}_a = [I_{aa}] \ddot{\boldsymbol{\phi}}_a + \dot{\boldsymbol{\phi}}_a^T [P_{aaa}] \dot{\boldsymbol{\phi}}_a + \mathbf{g}_a - [G_a^I]^T \mathbf{F}_{ext}, \tag{2}$$

where  $[I_{aa}]$  and  $[P_{aaa}]$  denotes the inertia matrix and inertial power array with respect to independent joint set, respectively.  $[G_a^I]$  is a Jacobian matrix that relates velocity at contact point to independent joints velocity.  $\mathbf{T}_a$ ,  $\mathbf{F}_{ext}$  and  $\mathbf{g}_a$  stands for joint torque vector, effective force vector, and gravity load, respectively. Since the position and velocities are finite during the impact, consequently the term involving integral  $\dot{\boldsymbol{\phi}}_a^T [P_{aaa}] \dot{\boldsymbol{\phi}}_a$  will be zero as  $\Delta t \rightarrow 0$ . Similarly, the terms involving actuation torque  $\mathbf{T}_a$  and gravity go to zero. Integration of (2) over contact time will yield [24]:

$$\Delta \boldsymbol{\phi}_a = [I_{aa}]^{-1} [G_a^I]^T \hat{\mathbf{F}}_{ext}. \tag{3}$$

where  $\hat{\mathbf{F}}_{ext} = \int_{t_0}^{t_0+\Delta t} \mathbf{F}_{ext} dt$  denotes the external impulse. The kinematic relationship between joint velocity and contact point velocity is established as follows:

$$[v_I] = [G_a^I] \dot{\boldsymbol{\phi}}_a. \tag{4}$$

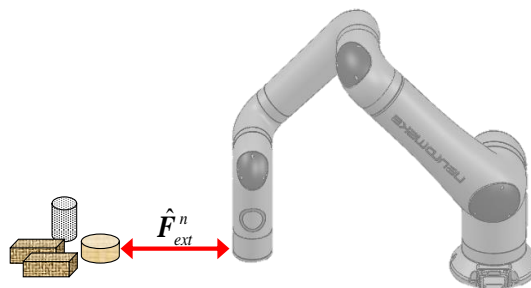


Figure 3. The scattering environmental model of robot and objects.

Finally, the velocity increment of the robot at contact point in terms of external impulses is established as:

$$[\Delta v_I] = [G_a^I] [I_{aa}]^{-1} [G_a^I]^T (\hat{\mathbf{F}}_{ext}), \tag{5}$$

Similarly, for the environment, the velocity increment relationship is established as:

$$[\Delta v_{env}] = [G_a^{env}] [I_{env}]^{-1} [G_a^{env}]^T (-\hat{\mathbf{F}}_{ext}). \tag{6}$$



Considering that there is no friction between the contacting surfaces, the impulse will always act along the normal vector  $\mathbf{n}$  at the contact point:

$$\hat{\mathbf{F}}_{ext} = \hat{F}_{ext} \mathbf{n}. \tag{7}$$

Now, by substituting Equations (5) and (6) into Equation (1), the closed form solution of the external impulse is established as follows:

$$\hat{\mathbf{F}}_{ext} = \frac{-(1+e)(\mathbf{v}_I - \mathbf{v}_{env})^T \mathbf{n}}{\mathbf{n}^T \{ [G_a^I] [I_{aa}]^{-1} [G_a^I]^T + [G_a^{env}] [I_{env}]^{-1} [G_a^{env}]^T \} \mathbf{n}}, \tag{8}$$

where the first term in the denominator is associated with the dynamics of the manipulator, while the second term shows the dynamic contribution of environment, respectively. During scattering, after the collision between robot and object, the object will collide with other objects in the environment. In next section, the collision model among objects is developed.

### 3.2. Collision Kinematics between Objects

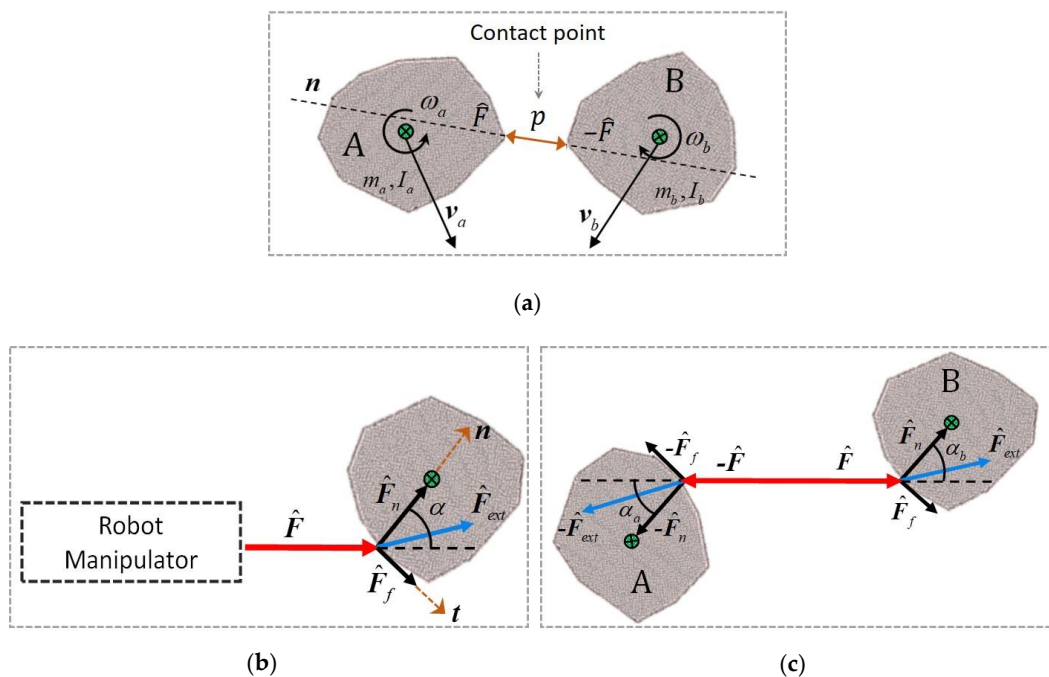
Consider two bodies with mass center velocities  $\mathbf{v}_a, \mathbf{v}_b$  and angular velocities  $\omega_a, \omega_b$  are approaching to each other as shown in Figure 4a. Where  $(m_a, I_a)$  and  $(m_b, I_b)$  denote the mass and mass moment of inertia of body 'A' and body 'B', respectively. During collision, impulsive force  $\hat{\mathbf{F}}_{ext}$  is experienced by both bodies with equal magnitude and in opposite direction. After collision, the mass center velocity of body 'A' can be expressed as follows:

$$\mathbf{v}'_a = \mathbf{v}_a + \hat{\mathbf{F}}_{ext} m_a^{-1}, \tag{9}$$

Similarly for body 'B':

$$\mathbf{v}'_b = \mathbf{v}_b - \hat{\mathbf{F}}_{ext} m_b^{-1}, \tag{10}$$

where  $\mathbf{v}'_a$  and  $\mathbf{v}'_b$  denote the post impact mass center velocities of body 'A' and body 'B', respectively.



**Figure 4.** (a) Collision model of two translating and rotating bodies. (b) Object and manipulator collision model by considering friction. (c) Collision model among objects by considering friction.

Similarly the post impact angular velocities can be written as follows:

$$\omega'_a = \omega_a + I_a^{-1}(\mathbf{r}_a \times \hat{\mathbf{F}}_{ext}) \tag{11}$$

and:

$$\omega'_b = \omega_b - I_b^{-1}(\mathbf{r}_b \times \hat{\mathbf{F}}_{ext}) \tag{12}$$

for body 'A' and body 'B', respectively. The contact point velocities of both bodies is written as follows:

$$\bar{\mathbf{v}}_a = \mathbf{v}_a + \omega_a \times \mathbf{r}_a, \tag{13}$$

and:

$$\bar{\mathbf{v}}_b = \mathbf{v}_b + \omega_b \times \mathbf{r}_b, \tag{14}$$

where  $\mathbf{r}_a$ , and  $\mathbf{r}_b$  denote the vectors directing from the center of the objects to the point of contact p.

Finally, the Equations (9)–(14) are rearranged to find the velocity increments of both bodies in terms of external as follows:

$$\Delta \mathbf{v}_a = \hat{\mathbf{F}}_{ext} m_a^{-1} + (\mathbf{r}_a \times \hat{\mathbf{F}}_{ext}) I_a^{-1} \times \mathbf{r}_a, \tag{15}$$

and:

$$\Delta \mathbf{v}_b = \hat{\mathbf{F}}_{ext} m_b^{-1} - (\mathbf{r}_b \times \hat{\mathbf{F}}_{ext}) I_b^{-1} \times \mathbf{r}_b, \tag{16}$$

where  $\Delta \mathbf{v}_a = \bar{\mathbf{v}}'_a - \bar{\mathbf{v}}_a$  and  $\Delta \mathbf{v}_b = \bar{\mathbf{v}}'_b - \bar{\mathbf{v}}_b$ .  $\bar{\mathbf{v}}'_a$  and  $\bar{\mathbf{v}}'_b$  denotes the post impact velocities at contact point of body 'A' and body 'B', respectively. Similar to Equation (8), the closed-form solution of normal impulse is obtained as follows:

$$\hat{\mathbf{F}}_{ext} = \frac{-(1 + e)(\bar{\mathbf{v}}_a - \bar{\mathbf{v}}_b) \cdot \mathbf{n}}{m_a^{-1} + m_b^{-1} + [(\mathbf{r}_a \times \mathbf{n}) I_a^{-1}] \times \mathbf{r}_a + (\mathbf{r}_b \times \mathbf{n}) I_b^{-1} \times \mathbf{r}_b] \cdot \mathbf{n}}. \tag{17}$$

In practical cases, for general shape objects, the impulsive force is resolved into two components: normal to the surface and tangential to the surface. For circular objects, if the friction coefficient  $\mu_f$  is negligible on the object surface, there will be only normal force directing toward the center of the objects. The Equations (8) and (17) denote this case. However, with consideration of the surface friction, an additional external impulse component will act along the tangential direction as shown in Figure 4b. For the case of Figure 4c,  $\hat{\mathbf{F}}_n$  and  $\hat{\mathbf{F}}_t$  are normal and tangential component of impulse experienced by the object.  $\hat{\mathbf{F}}_f$  is the frictional impulse, which is associated with  $\mu_f \hat{\mathbf{F}}_n$ . The slipping phenomenon will happen if  $\hat{\mathbf{F}}_t$  is greater than  $\mu_f \hat{\mathbf{F}}_n$ . The total external impulse experienced by an object will be a vector sum of normal and tangential components, whose magnitude is given as follows:

$$\hat{\mathbf{F}}_{ext} = \sqrt{\hat{\mathbf{F}}_f^2 + \hat{\mathbf{F}}_n^2}. \tag{18}$$

The object experiences impulsive torque due to the frictional external component  $\hat{\mathbf{F}}_f$ , which is given as follows [25]:

$$\hat{\boldsymbol{\tau}} = \mathbf{r}_a \times \hat{\mathbf{F}}_f = [I_a] \Delta \omega_a, \tag{19}$$

where  $\Delta \omega_a$  denotes the change in angular velocity of the object. If  $\hat{\mathbf{F}}_t$  impulse component is smaller than  $\mu_f \hat{\mathbf{F}}_n$ , then will be no slip and accordingly we have:

$$\hat{\mathbf{F}}_f = \hat{\mathbf{F}}_t, \tag{20}$$

where the tangential component is given as follows:

$$\hat{\mathbf{F}}_t = \hat{\mathbf{F}} - \hat{\mathbf{F}}_n. \tag{21}$$

However, if  $\hat{F}_t$  is larger than  $\mu_f \hat{F}_n$  then slip occurs and the amount of impulse transmitted along tangential direction will be:

$$|\hat{F}_f| = |\mu_f \hat{F}_n| t, \quad (22)$$

where:

$$|\hat{F}_n| = |\hat{F}| \cos \alpha, \quad (23)$$

### 3.3. Condition for Scattering and Pushing

In this section, the condition for scattering and pushing are stated. During collision, due to actuation of the manipulator, the change in velocity of the manipulator is small compared to change in velocity of the object. Consider that  $\Delta v_I \ll \Delta v_{inv}$ , Equation (8) will be reduced to:

$$\hat{F}_{ext} = -(1 + e)m_a(v_I - v_a). \quad (24)$$

which implies that the velocity of the manipulator before and after collision will remain the same ( $v'_I = v_I$ ), where  $v_I$  and  $v'_I$  denotes the velocities of the manipulator before and after collision. The velocity of the object after collision is given as follows:

$$v'_a(t_0) = v_a(t_0) - (1 + e)v_I. \quad (25)$$

where  $v_a(t_0)$  denotes the initial velocity of the object at  $t_0$  before collision which is always zero in our application.  $v'_a(t_0)$  denotes the initial velocity of the object after collision with the robot manipulator.

Finally, the object will come to rest with the following velocity:

$$v'_a(t) = v'_a(t_0) - \text{sign}(v)\mu_a g t \quad (26)$$

The condition for scattering and pushing is associated with relative velocities of robot manipulator and object after collision. For the case of pushing (quasi-static manipulation), the condition states that:

$$v'_I \geq v_a(t_0), \text{ Object sticking} \quad (27)$$

Similarly for the case of scattering (dynamic manipulation), the condition will be formed as follows:

$$v'_I < v_a(t_0). \text{ Object sliding apart} \quad (28)$$

For the case of pushing (quasi-static manipulation), the object motion will be determined by the motion of the manipulator only. Accordingly, the dynamics of the object will not affect the motion of the object. This paper deals solely with the dynamic manipulation (scattering) of the objects, where the motion of the object is determined by the motion of the manipulator and dynamics of the object. Accordingly, the dynamics of the object, coefficient of friction, and coefficient of restitutions should be known to estimate the final position of the objects.

## 4. Learning Physical Properties

The experimental setup is shown in Figure 5. A 4K (Sony FDR-AX30, Seoul, Korea) camera with 60-frame rate is used to record the videos. For scattering, an Indy 7 Neuromeka robotic arm is used (task space velocity ranges from 0 to 1 m/s). Six objects with different shapes, sizes, and masses are used during training and experimentation. In this section, the physical properties of objects (i.e., friction coefficient, coefficient of restitution, mass ratio of object) are estimated from the observed parameters, such as travel distance, velocity, and acceleration of the object. The parameters are observed through dynamic interaction of objects. A black marker is placed on each object and videos are recorded by colliding a single object with the manipulator and two objects.

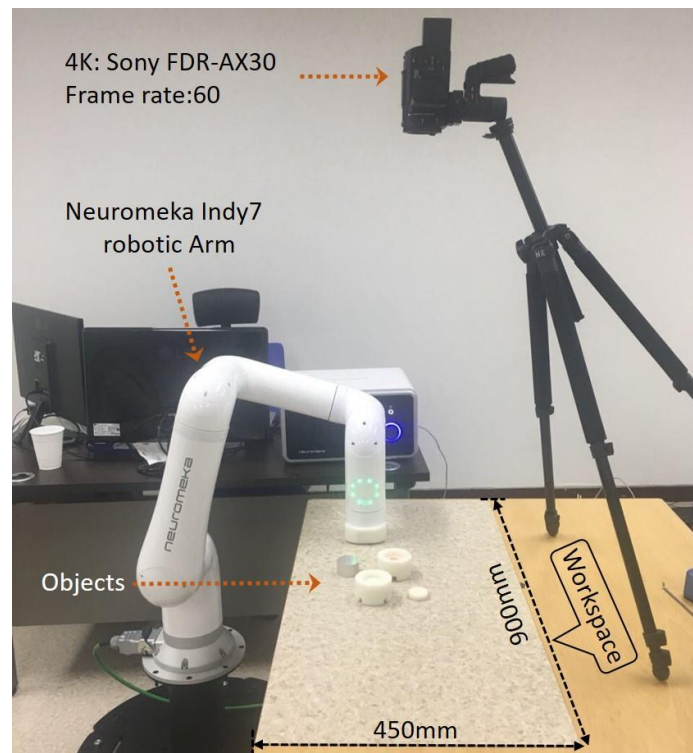


Figure 5. Experimental setup for video collection.

Video images are converted into binary images and all connected components are labeled using the ‘8-connectivity’ method. Furthermore, the noise and the undesired connected region are removed. Finally, to find the centroid of objects, blob analysis is applied to detect the connected objects in the frame by defining the minimum and maximum blob area. Once the frame rate and centroid of the objects are known in each frame, we can find the travel distances, velocities, and accelerations of the objects from the videos. Figure 6 illustrates the observed parameter from video, when two objects (A and C) collide with manipulator. At  $t = 1.5$  s (first collision), the robot collides with object A and at  $t = 1.68$  s (second collision), the object A collide with the object C.

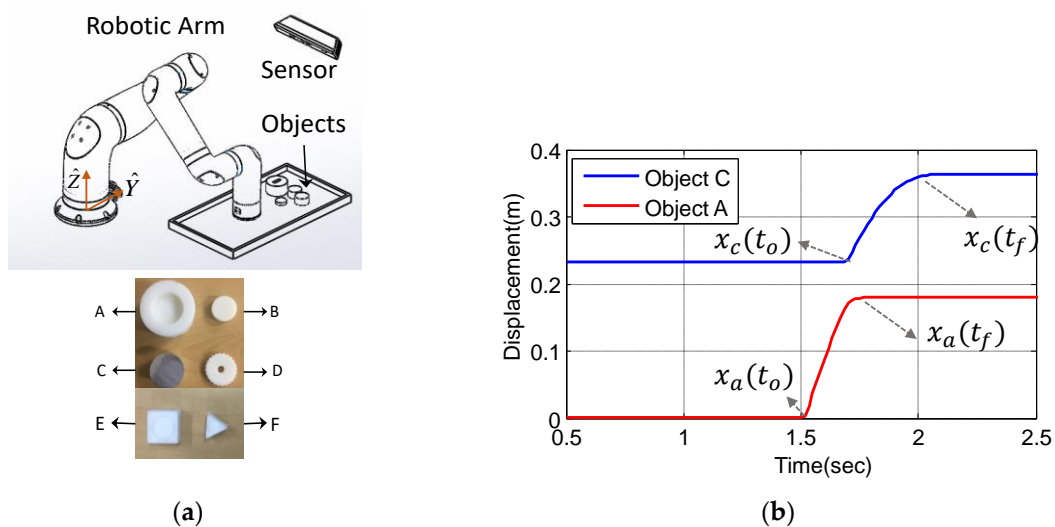
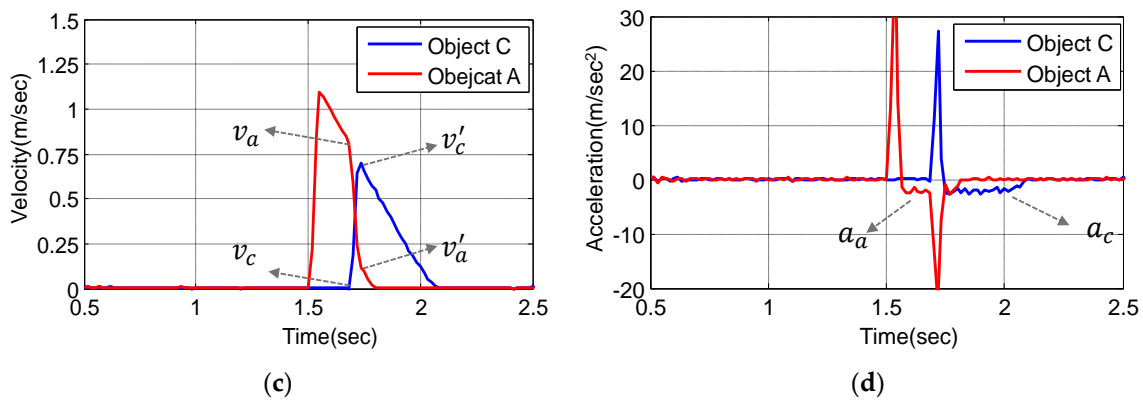


Figure 6. Cont.



**Figure 6.** Setup, objects, and observed parameters from collision of two objects. (a) Setup and object involved in scattering application. (b) Observed displacements. (c) Observed velocities. (d) Observed accelerations.

#### 4.1. Coefficient of Friction

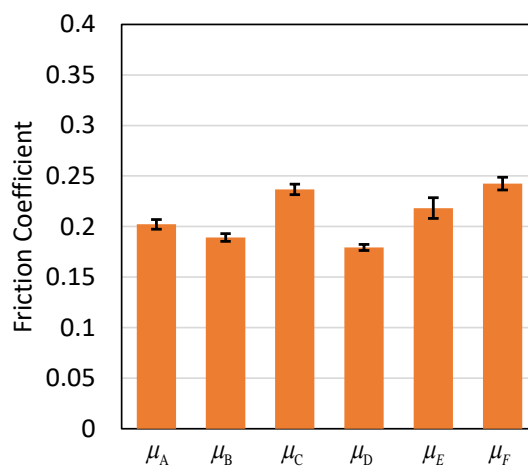
Assume that an object with mass  $m$  moves with acceleration  $a$  on table surface with coefficient of friction  $\mu$ . For sliding objects, the direction of the friction is opposite to the direction of the velocity of the object. Accordingly,  $ma = -sign(v).m\mu g$  holds. The coefficient of friction between each object and surface of table is expressed either by:

$$a = -sign(v).\mu g, \tag{29}$$

or as its integrated form:

$$x(t) = x(t_0) + v(t_0)t - \frac{1}{2}sign(v).\mu gt^2. \tag{30}$$

The distance travelled by objects and acceleration after collision is shown in Figure 6b,d, respectively. Five videos are taken for each object by creating collisions with the manipulator. The coefficients of frictions are calculated by using Equation (29). The average value of the coefficient of friction, with standard deviation, is given in Figure 7.



**Figure 7.** Physical properties. Average value and standard deviation of coefficient of friction of four objects shown in Figure 5a.

#### 4.2. Coefficient of Restitution

A coefficient of restitution is associated with type of material of two objects involved in a collision. During the scattering process, after an object is hit by the manipulator, it collides with other objects in

the environment. Furthermore, the impulse applied by the manipulator is propagated to all objects involved in the collision. To model all collisions, the value of  $e$  (coefficient of restitution) between the manipulator and each object and between each object and all other objects should be known. The colliding velocity of the robot manipulator is known. Furthermore, by using the observable parameters (velocity of the object, before and after collision) from the video images, the value of  $(e)$  between manipulator and each object is calculated from the following relation:

$$v_a' - v_a = -\frac{\hat{F}_{ext}}{m_a} = -(1 + e)v_I \tag{31}$$

Next, the value of coefficient of restitution ( $e$ ) between objects is calculated. The velocities of objects before and after collision was observed from videos by colliding each object with all other objects (one by one) as shown in Figure 6c. Based on these observed velocities, the coefficient of restitution between objects A and B is calculated as follows:

$$e = \frac{(v'_a - v'_b)}{(v_b - v_a)} \tag{32}$$

Similarly, the value of  $(e)$  is estimated among other objects and. For estimation of  $(e)$ , five videos of each scenario were recorded. The average values of estimated coefficient of restitutions with standard deviation are shown in Figure 8.

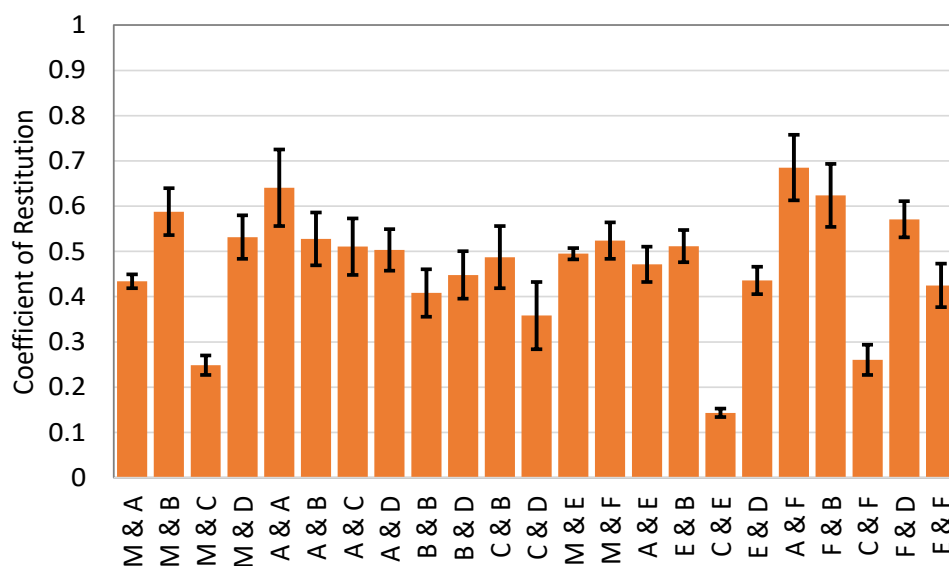


Figure 8. Coefficient of restitution: Average value and standard deviation. M stands for manipulator.

### 4.3. Mass Ratio of Objects

In this section, analytical methodology is provided to estimate the masses of the objects. In dynamic manipulation, the motion of the objects is determined based on both the dynamic characteristic of the objects and motion of the manipulator. Accordingly, it is important to infer mass properties of the objects to model their motion after impulsive actions. Like other physical properties, the masses ratio of objects are also estimated based on observed parameters and combining with analytical model. Substitution of Equation (17) into Equation (15) will yield:

$$\Delta v_a = \frac{-(1+e)(\bar{v}_a - \bar{v}_b) \cdot n}{1 + \frac{m_a}{m_b} + 2[(r_a \times n)^2 / r_a^2 + (r_b \times n)^2 / r_b^2] \cdot \frac{m_a}{m_b}} \cdot n + r_a \times \frac{-(1+e)(\bar{v}_a - \bar{v}_b) \cdot n}{2r_a^2 \left(1 + \frac{m_a}{m_b}\right) + [(r_a \times n)^2 + (r_b \times n)^2] \cdot \frac{m_a}{m_b}} \cdot n \times r_a \tag{33}$$



Similarly, the substitution of Equation (17) into Equation (16) will yield:

$$\Delta v_b = \frac{-(1+e)(\bar{v}_a - \bar{v}_b) \cdot \mathbf{n}}{1 + \frac{m_b}{m_a} + 2[(r_a \times \mathbf{n})^2 / r_a^2 \cdot \frac{m_b}{m_a} + (r_b \times \mathbf{n})^2 / r_b^2]} \cdot \mathbf{n} - \tag{34}$$

$$r_b \times \frac{-(1+e)(\bar{v}_a - \bar{v}_b) \cdot \mathbf{n}}{2r_b^{-2}(1 + \frac{m_b}{m_a}) + [(r_a \times \mathbf{n})^2 / r_a^2 \cdot \frac{m_b}{m_a} + (r_b \times \mathbf{n})^2]} \cdot \mathbf{n} \times r_b,$$

where all parameters except the ratio of masses ( $\frac{m_b}{m_a}, \frac{m_b}{m_a}$ ) are already known. The velocities before/after collision are observed from the videos and the coefficient of restitutions are already estimated. Thus, it is inferred from the above equations that the change in velocities of two objects being involved in collision is associated with ratio of masses of two objects. That is, there exists infinite solutions for  $m_a$  and  $m_b$ , which implies that the equations can be written in form of  $[x \ y] \begin{pmatrix} m_a \\ m_b \end{pmatrix} = 0$ , where  $x$  and  $y$  are known values. It is noted that the ratio of the masses is always the same no matter what equation is analyzed. That solution will be enough to model the object’s trajectory after collision even though the true masses are unknown. Figure 9 denotes the average values and standard deviation of mass ratio of each object relative to other objects.

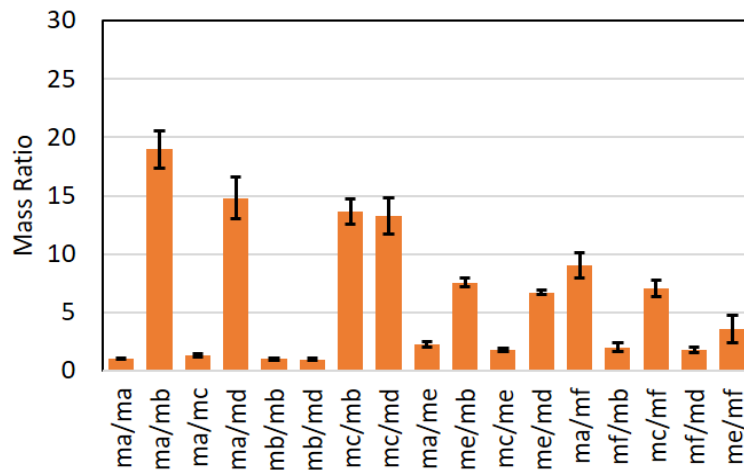


Figure 9. Physical properties. Average value and standard deviation of mass ratio of objects.

#### 4.4. Exactness of Estimated Parameters

The accuracy of a virtual world simulator is directly related to the estimation of physical properties. Accordingly, it is critical to analyze precisely, how these physical properties are estimated. Practically, the ground truth of frictional coefficient varies even on one surface, which makes the impulsive actions difficult to control the moving objects on an exact location. However, for object singulation application, the impulsive actions can still provide useful motions as the target kinematic constraint are not so restrictive. Except the true value of masses of all objects (i.e.,  $m_a = 143g$ ,  $m_b = 8g$ ,  $m_c = 112g$ ,  $m_d = 9g$ ,  $m_e = 61g$ ,  $m_f = 16g$ ), the true values of two other estimated physical properties such as coefficient of friction and coefficient of restitution, are unknown. However, the coefficient of restitution is being used while estimating the mass ratios of objects as given in Equations (33) and (34). Accordingly, the error between estimated and real values of the mass ratio depicts the trueness of the mass ratio and coefficient of restitutions of objects. The percentage error between real/estimated mass ratios is less than 10% as calculated in Table 1. Accordingly, it is inferred that the estimated mass ratios and coefficient of restitutions are precise enough to make virtual world for scattering application. The exactness of the coefficient of friction and coefficient of restitutions between object and manipulator can be analyzed based on the real-time performance of the proposed methodology. In Section 6, the results are quantitatively compared by measuring the

distance among objects in real world and virtual world, after the collision. The error among distances depicts the exactness of estimated coefficient of friction.

**Table 1.** Percentage error between mass ratio of the true value and the estimated value.

Objects	ma/ma	ma/mb	ma/mc	ma/md	mb/mb	mb/md	mc/mb	mc/md	ma/me
Error %	3.47	6.15	4.8	6.9	1.7	10	2.5	6.5	3.13
Objects	me/mb	mc/me	me/md	ma/mf	mf/mb	mc/mf	mf/md	me/mf	
Error %	7.7	2.91	0.79	8.75	5.98	1.05	3.14	6.17	

### 5. Scattering Algorithm and Experimentation

In this section, the developed analytical methodology is verified by performing the scattering experiments. We use six different objects with distinct physical properties, such as mass, coefficient of restitution, coefficient of friction, and shape, for object singulation experiment. The objects singulation is performed based on the proposed scattering algorithm. Once all the physical properties of the objects are identified, we can integrate them in a physics engine to design a virtual world simulator to perform controlled scattering. We conducted scattering experiments in the virtual world to scatter objects in a desired manner. First of all, it is necessary to define a scattering index that can be used as a measure of scattering performance. Here, we need to provide some constraints to the scattering task:

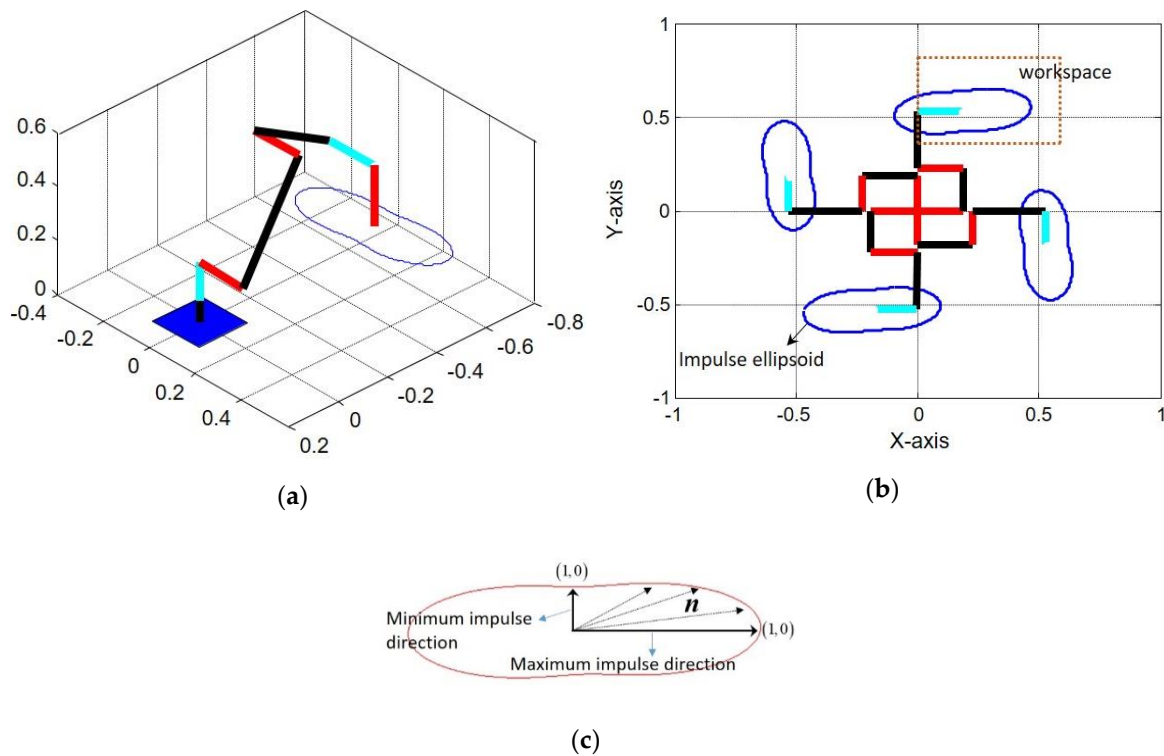
- (i) Initially, we assumed that all objects are placed together.
- (ii) The distance among objects after scattering does not go beyond the given workspace. Maximum and minimum distances ( $S_{min}$ ,  $S_{max}$ ) are given.
- (iii) Hitting strategy: Among objects, the first priority is to hit the largest object in the cluster for an initial collision to scatter all objects. The reason for hitting the largest object is to provide enough impulse to drive out other smaller objects.
- (iv) How to hit the largest object: This is based on a central impact to simplify the number of parameters. Other than that, the number of parameters needed to estimate the physical properties are increased.
- (v) Magnitude of the velocity: In terms of deciding the magnitude of colliding velocity, it is based on the scattered distances among objects. We conducted simulations several times until all objects satisfied the condition of scattering.
- (vi) Direction of the robot motion: The direction of robot motion is selected considering the ability of the manipulator to generate impulse in certain direction. Kim et al. [26] proposed the normalized impact geometry to analyze the impulse generation ability of the manipulator. Figure 10 shows the generalized impulse geometry for the robot manipulator, which is constructed based on the impulse model of Equation (8). Figure 10b,c shows the maximum and minimum impulse direction for four different configurations. The direction of robot motion is decided to apply the maximum impulse. However, in some cases the largest object is surrounded by other objects in clutter and not accessible for initial collision with the robot manipulator. In this scenario, the direction of robot motion is decided to maximize the external impulse without considering the constraint (iii).
- (vii) The scattering index is set as the average distance among objects. It can be expressed as:

$$s_{min} \leq \sum_{i=1}^M \sum_{j=i+1}^M \frac{\|r_i - r_j\|}{N} < s_{max}, \text{ where } N = {}_M C_2, \tag{35}$$

subject to the distance constraint between objects  $i$  and  $j$ :

$$\|r_i - r_j\| \geq d_{min}(i \neq j), \tag{36}$$

where  $r_i$  is the mass center of  $i^{th}$  object in plane.  $N$  and  $M$  are the number of objects and the number of connection of objects, respectively. Equation (35) implies the average distance among all objects and Equation (36) implies that distance between two consecutive objects should be greater or equal to  $d_{min}$ . The additional constraint of Equation (36) is included to make sure that all objects are apart from each other.



**Figure 10.** Impulse geometry: (a) normalized impulse ellipsoid for the Indy 7 robotics arm; (b) impulse ellipsoid for the four configurations; (c) maximum and minimum impulse direction.

Identification and visual segmentation of objects in a cluttered environment were important research issues. However, thanks to recent developments in deep learning techniques, the performance of semantic segmentation and instance segmentation algorithms [27–29] has improved significantly. Moreover, some of them provide open source code [30] to help other researchers to train their own datasets. Therefore, with the techniques mentioned above, it is straightforward to identify and segment objects in an image for robotic grasp. However, since this paper concentrated solely on the mechanic issue, we are currently assuming that locations of all parts are already known. Considering the recent development of segmentation, we believe that our methodology can be easily extended to scenarios that require object identification and segmentation processes.

The scattering algorithm is explained in Figure 11. The initial positions of all objects are obtained from the real world and are fed to the virtual world. Then, the scattering experiment is performed in the virtual world until the desired distances among objects are achieved as defined in Equations (35) and (36). DAFUL [31] software, made of Virtual Motion Co., was employed in the virtual world simulations. Here, a feedback routine is employed to minimize the error between the desired distance and current distance by updating the colliding velocity of the robot manipulator. Once the error is within the allowed range, the objects are singulated. Finally, that same desired velocity will be given as the input to the robot manipulator and real-time scattering is performed. During experimentation, the minimum distance among objects is set as  $d_{min} = 90$  mm. The average maximum and minimum maximum distance is set as  $S_{min} = 150$  mm,  $S_{max} = 900$  mm. Once all physical parameters are estimated, based on the ratio of masses and coefficient of restitutions, the change in velocity is calculated by using

Equations (33) and (34). Finally, Equation (30) is employed to calculate the travel distance of each object after collision. All estimated physical parameters will be used in this process. The scattering results between two objects are shown in Figure 12. Figure 12 shows two scenes; one from the virtual world and the other from the real world. Figure 12b shows the results after scattering. It was found from scattering experiment that the objects are singulated after controlled scattering and the results of singulation are similar in both cases. Figures 13 and 14 demonstrate the case of three objects by considering the circular and general shapes, respectively.

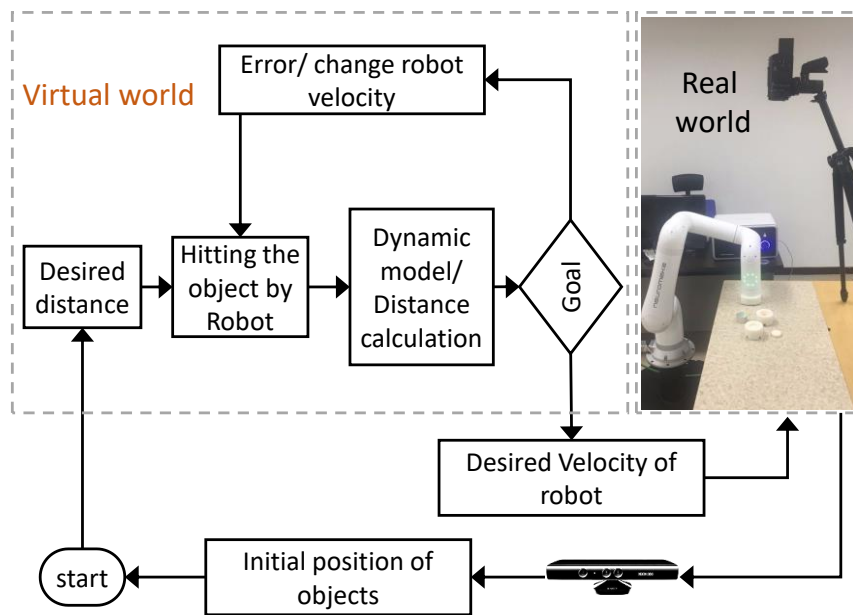


Figure 11. Scattering algorithm for object singulation.

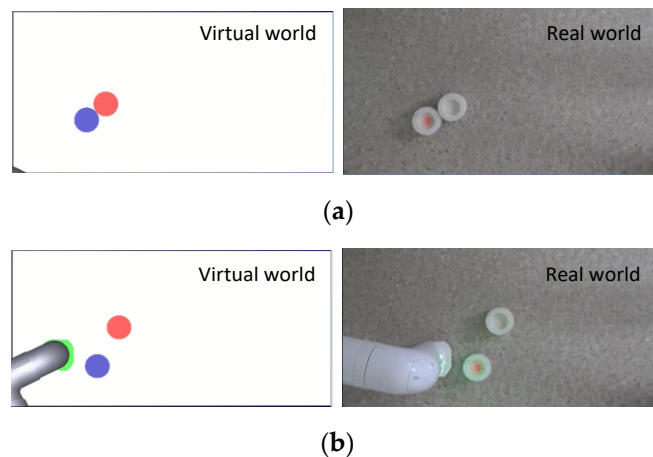
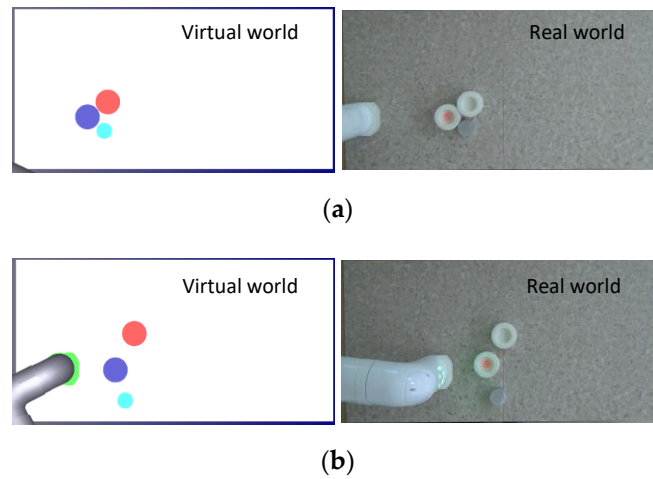
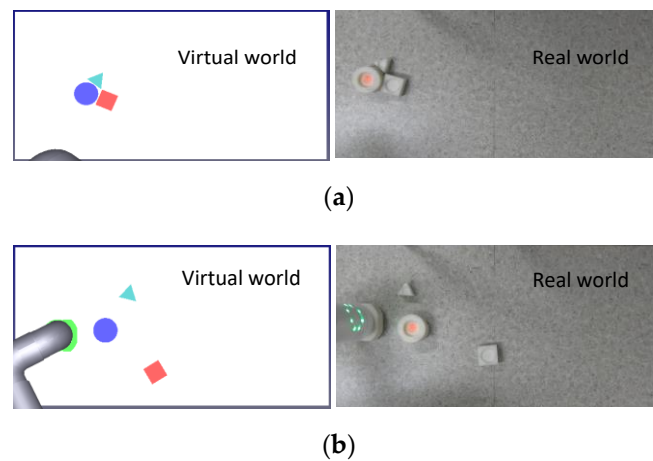


Figure 12. Scattering experiments for singulation of two objects: (a) before collision; (b) after collision. Manipulator velocity: 0.4 m/s.

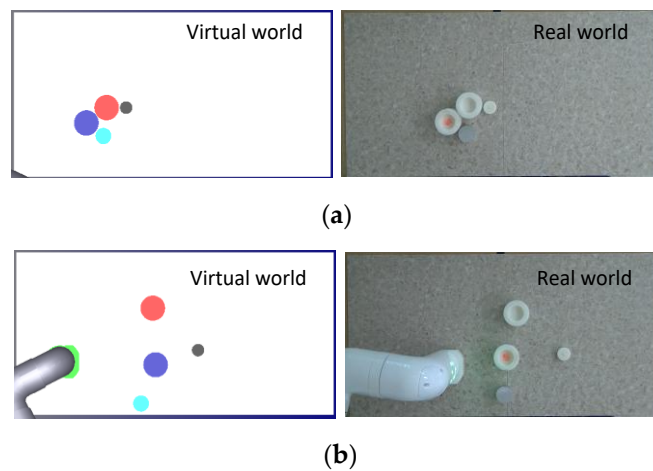
Finally, the proposed technique was successfully applied and verified by considering the four objects in the environment. Figures 15 and 16 show the results of singulation in real and virtual world for circular shape and general shape objects, respectively. As mentioned before, the impulsive actions-based dynamic manipulation of objects makes it difficult to locate an object exactly on target. However, it is inferred from Figures 12–16 that the dynamic manipulation can be successfully applied for object singulation application and performance of virtual world and successful implementation in the real world is satisfactory.



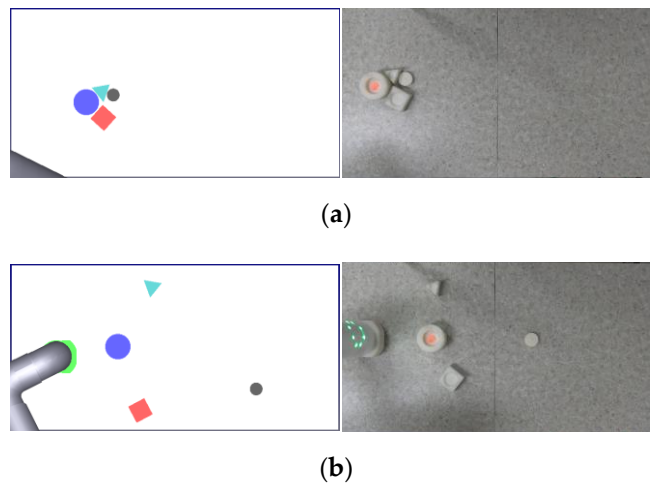
**Figure 13.** Scattering experiments with three objects by considering circular shapes only: (a) before collision; (b) after collision. Manipulator velocity: 0.5 m/s.



**Figure 14.** Scattering experiments for singulation of three objects by considering general shapes: (a) before collision; (b) after collision. Manipulator velocity: 0.4 m/s.



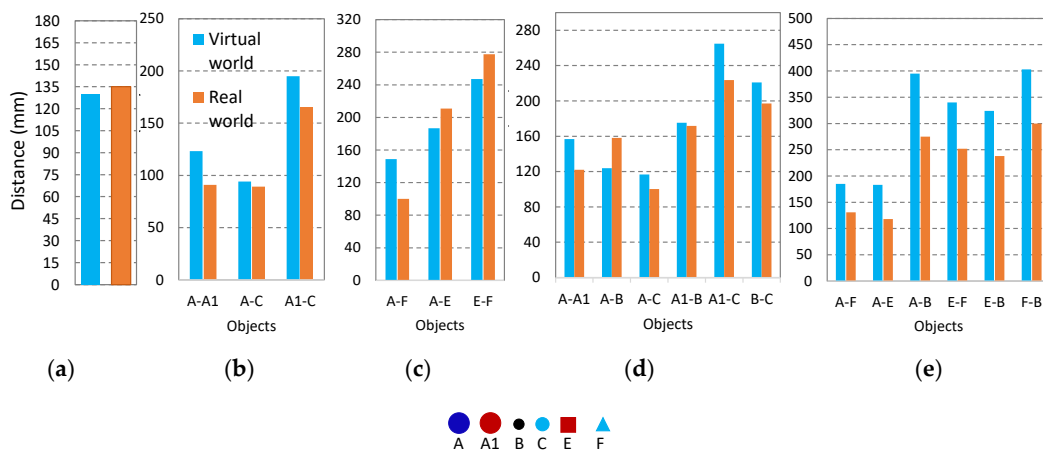
**Figure 15.** Scattering experiments for singulation of four objects by considering circular shapes only: (a) before collision; (b) after collision. Manipulator velocity: 0.6 m/s.



**Figure 16.** Scattering experiments for singulation of four objects by considering general shapes: (a) before collision; (b) after collision. Manipulator velocity: 0.5 m/s.

### 6. Quantitative Analysis

In this section, the real and virtual world results are compared to evaluate how exactly the physical properties are estimated. We measured the distance among all objects after singulation in the virtual world and real world. The quantitative data is shown in Figure 17. Figure 17a–c denote the comparison of real/virtual world’s distance for the case of two objects and three objects singulation, respectively. Similarly, the distance comparison of real/virtual world for the case of four objects is shown in Figure 17d,e. There is a small error, which is expected because we are using the average value of estimated parameters, and ground truth of the friction coefficient varies even on one surface [13]. The error is comparatively grater for the case of general shape objects. This is because, for circular shaped objects, the contact direction is always toward the center. It is confirmed that the distance between any two objects is greater than  $d_{\min} = 90$  mm and that the condition given in Equation (35) is also satisfied in all cases. Conclusively, the proposed scattering algorithm was found to be effective to analyze and control the scattering behavior of objects being singulated by collision with a robot manipulator. The Supplementary Material (entitle Singulation of objects in virtual and real world) is a video clip that demonstrates the whole procedure of training the physical properties of objects and scattering experiments.



**Figure 17.** The quantitative analysis between real world and virtual world. A comparison between real/virtual world distances among all objects. (a) Two-object singulation; (b) three-object singulation by considering the circular shape only; (c) three-object singulation by considering the general shape of objects; (d) four-object singulation considering the circular shape only; (e) four-object singulation considering the general shape of objects.



## 7. Conclusions

In this paper, a scattering technique for object singulation in a cluttered environment is proposed. The main idea was to design a virtual world simulator based on the estimation of physical properties of the environment. In order to obtain physical properties of the objects, an impulse-based approach was used along with image processing technique to acquire observable parameters from videos. A virtual world simulator was employed to perform scattering in a desired manner. Finally, the effectiveness of the proposed scattering algorithm was verified through real-world scattering experiments.

It will be difficult to perform model-based object singulation for complex-shaped objects with a non-uniform pressure distribution. Our future aim is to combine the analytical model with deep learning. The neural networks will be trained based on the simulations of a virtual world for the general shape of objects (circular, triangular, cube). By performing enough training of scattering patterns, we can perform scattering of complex-shaped objects in the virtual world, which are not being used during training. Finally, the virtual world input will be fed to the robot to perform real-world object singulation.

Another ongoing work is developing an analytical model to distinguish pushing and scattering. In the real world, objects are singulated through physical contact between the object and manipulator. However, due to conditions of contact, such as the strength and speed of interaction, material properties, or environment friction, contact between the object and manipulator may be maintained or not. If contact is maintained during motion, it is the case of pushing (quasi-static manipulation). Other than that, the resulting behavior will be complete separation, as in the case of scattering (dynamic manipulation). Developing a relevant analytical model would facilitate analysis and control singulation of objects.

**Supplementary Materials:** The following are available online at <http://www.mdpi.com/2076-3417/9/17/3536/s1>, Video S1: Singulation of objects in virtual and real world.

**Author Contributions:** Conceptualization: I.H.S. and B.-J.Y.; methodology: A.I. and B.-J.Y.; software: A.I. and S.-H.K.; writing—original draft preparation: A.I.; writing—review and editing: I.H.S. and Y.-B.P.; supervision: B.-J.Y.; experimentation: S.-H.K. and A.I.

**Funding:** This research was funded by the Technology Innovation Program (or Industrial Strategic Technology Development Program) (grant number 20001856, Development of Robotic Work Control Technology Capable of Grasping and Manipulating Various Objects in Everyday Life Environment Based on Multimodal Recognition and Using Tools) funded By the Ministry of Trade, Industry and Energy (MOTIE, Korea), by the Ministry of Trade, Industry and Energy (MI, Korea), and performed by the ICT-based Medical Robotic Systems Team of Hanyang University, Department of Electronic Systems Engineering was supported by the BK21 Plus Program funded by National Research Foundation of Korea (NRF). The APC was funded by grant number 20001856.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Caldera, S.; Rassau, A.; Chai, D. Review of deep learning methods in robotic grasp detection. *Multimodal Technol. Interact.* **2018**, *2*, 57. [[CrossRef](#)]
2. Sarantopoulos, I.; Doulgeri, Z. Human-inspired robotic grasping of flat objects. *Robot. Auton. Syst.* **2018**, *108*, 179–191. [[CrossRef](#)]
3. Argall, B.D.; Chernova, S.; Veloso, M.; Browning, B. A survey of robot learning from demonstration. *Robot. Auton. Syst.* **2009**, *57*, 469–483. [[CrossRef](#)]
4. Laskey, M.; Lee, J.; Chuck, C.; Gealy, D.; Hsieh, W.; Pokorny, F.T.; Dragan, A.D.; Goldberg, K. Robot grasping in clutter: Using a hierarchy of supervisors for learning from demonstrations. In Proceedings of the IEEE Conference on Automation Science and Engineering, Fort Worth, TX, USA, 21–25 August 2016; pp. 827–834.
5. Ross, S.; Bagnell, D. Efficient reductions for imitation learning. In Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010; pp. 661–668.
6. Kloss, A.; Schaal, S.; Bohg, J. Combining learned and analytical models for predicting action effects. *arXiv* **2017**, arXiv:1710.04102.

7. Eitel, A.; Hauff, N.; Burgard, W. Learning to singulate objects using a push proposal network. *arXiv* **2017**, arXiv:1707.08101.
8. Hermans, T.; Rehg, J.; Bobick, A. Guided pushing for object singulation. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and System, Vilamoura, Portugal, 7–12 October 2012; pp. 4783–4790.
9. Chang, L.; Smith, J.R.; Fox, D. Interactive singulation of objects from a pile. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN, USA, 14–18 May 2012; pp. 3875–3882.
10. Dogar, M.; Hsiao, K.; Ciocarlie, M.; Srinivasa, S. Physics-Based grasp planning through clutter. *Robot. Sci. Syst.* **2012**, *2012*, 78–85.
11. Dogar, M.; Srinivasa, S. Push-grasping with dexterous hands: Mechanics and a method. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; pp. 2123–2130.
12. Zhou, J.; Paolini, R.; Bagnell, J.A.; Mason, M.T. A convex polynomial force-motion model for planar sliding: Identification and application. In Proceedings of the IEEE International Conference on Robotics and Automation, Stockholm, Sweden, 16–21 May 2016; pp. 372–377.
13. Yu, K.T.; Bauza, M.; Fazeli, N.; Rodriguez, A. More than a million ways to be pushed a high-fidelity experimental dataset of planar pushing. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Daejeon, South Korea, 9–14 October 2016; pp. 30–37.
14. Bicchi, A.; Kumar, V. Robotic grasping and contact: A review. In Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, CA, USA, 24–28 April 2000; pp. 348–353.
15. Wu, J.; Lim, J.; Zhang, H.; Tenenbaum, J.; Freeman, W.; Wilson, R.C.; Hancock, E.R.; Smith, W.A.P.; Pears, N.E.; Bors, A.G. Physics 101: Learning physical object properties from unlabeled videos. *Br. Mach. Vis. Conf.* **2016**, *2016*, 2.
16. Wu, J.; Yildirim, I.; Lim, J.J.; Freeman, B.; Tenenbaum, J.; Cortes, C.; Lawrence, N.D.; Lee, D.D.; Sugiyama, M.; Garnett, R. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. In Proceedings of the Advances in Neural Information Processing Systems 28 (NIPS 2015), Montreal, Canada, 7–12 December 2015; pp. 127–135.
17. Walker, I.D. Impact configurations and measures for kinematically redundant and multiple armed robot systems. *IEEE Trans. Robot. Autom.* **1994**, *10*, 670–683. [[CrossRef](#)]
18. Imran, A.; Yi, B.-J. Impulse modeling and new impulse measure for human-like closed-chain manipulator. *IEEE Robot. Autom. Lett.* **2016**, *1*, 868–875. [[CrossRef](#)]
19. Imran, A.; Yi, B.-J. Motion optimization of human body for impulse-based applications. *Intell. Serv. Robot.* **2018**, *11*, 323–333. [[CrossRef](#)]
20. Barghijand, H.; Akbarimajd, A.; Keighobadi, J. Quasi-Static object manipulation by mobile robot: Optimal motion planning using GA. In Proceedings of the International Conference on Intelligent Systems Design and Applications, Cordoba, Spain, 22–24 November 2011; pp. 202–207.
21. Lynch, K.M. Locally controllable polygons by stable pushing. In Proceedings of the IEEE International Conference on Robotics and Automation, Albuquerque, NM, USA, 25 April 1997; pp. 1442–1447.
22. Goyal, S.; Ruina, A.; Papadopoulos, J. Planar sliding with dry friction. Parts 1: Limit surface and moment function. *Wear* **1991**, *143*, 307–330. [[CrossRef](#)]
23. Wittenburg, J.; Likins, P. *Dynamics of Systems of Rigid Bodies*; B. G. Teubner: Stuttgart, Germany, 1978.
24. Imran, A.; Yi, B.-J. A closed-form analytical modeling of internal impulses with application to dynamic machining task: Biologically inspired dual-arm robotic approach. *IEEE Robot. Autom. Lett.* **2018**, *3*, 442–449. [[CrossRef](#)]
25. Choi, J.Y.; Yi, B.-J. Dynamics and impact control of a flying soccer ball. *J. Korean Phys. Soc.* **2009**, *54*, 75–84. [[CrossRef](#)]
26. Kim, J.; Chung, W.K.; Youm, Y. Normalized impact geometry and performance index for redundant manipulators. In Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, CA, USA, 24–28 April 2000; pp. 1714–1719.
27. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [[CrossRef](#)] [[PubMed](#)]

28. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [[CrossRef](#)] [[PubMed](#)]
29. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
30. Detectron. Available online: <https://github.com/facebookresearch/Detectron> (accessed on 1 December 2018).
31. Virtual Motion. Available online: <http://www.virtualmotion.co.kr/index.do> (accessed on 31 July 2019).



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).