Article

Scattering or Pushing for Object Singulation in Cluttered Environment: Case Study with Soma Cube

Muhammad Umair Ahmad Khan [1], Abid Imran [2], Sanghwa Kim [3], Hyunho Hwang [3], Ji Yeong Lee [4],*, Sungon Lee [3] and Byung-Ju Yi [5],*

[1] Department of Mechatronics Engineering, Hanyang University, Ansan 15588, Korea; umairsiyal@gmail.com

[2] Faculty of Mechanical Engineering, GIK Institute of Engineering Sciences and Technology, Swabi 23640, Pakistan; abid_mecha@yahoo.com

[3] Department of Electrical and Electronic Engineering, Hanyang University, Ansan 15588, Korea; sonata7943@hanyang.ac.kr (S.K.); huno10239@gmail.com (H.H.); sungon@hanyang.ac.kr (S.L.)

[4] Department of Robotics Engineering, Hanyang University, Ansan 15588, Korea

[5] Department of Electronic and Electrical Engineering, Hanyang University, Ansan 15588, Korea

* Correspondence: jiyeongl@hanyang.ac.kr (J.Y.L.); bj@hanyang.ac.kr (B.-J.Y.); Tel.: +82-10-8819-7229 (J.Y.L.); +82-10-2065-5218 (B.-J.Y.)

Featured Application: Robotic grasping and manipulation.

Abstract: This paper proposes a hybrid singulation approach combining dynamic scattering and pushing techniques. Using the impulse-based dynamic model, a guideline is provided to decide whether scattering or pushing is conducted for object singulation. The Soma cube consisting of seven blocks is used as the test example. The target is to singulate all the blocks. The dynamic scattering technique was initially applied to separate blocks in the formation. However, scattering alone does not provide target singulation in all the cases. So we combine the quasi-static pushing technique to complete the singulation of all the blocks. In pushing, image segmentation based on principal component analysis (PCA) algorithm was employed to singulate multiple blocks in clutter and prehensile manipulation was used to remove isolated blocks. Several 2-D formations of the Soma cube are used as the test cases. To validate the effectiveness of our approach, we have conducted comparative analysis which clearly shows that the hybrid singulation achieves singulation in much less time as compared to the pure pushing approach.

1. Introduction

Picking up objects among clutter seems easy for humans because the human brain instantly and intuitively determines different ways to complete the task. However, even simple tasks are more complicated for robots. Assume that many objects are cluttered in the environment. Using non-prehensile manipulation, there are two ways to manipulate objects: quasi-static and dynamic manipulation methods. In case of quasi-static manipulation (i.e., pushing for object singulation), there is always a contact between the object and the manipulator. Usually, the quasi-static manipulation consumes time and as a result we need multiple pushing to get the desired singulation between objects. Pushing approach to grasp ungraspable objects inspired by human strategies has been surveyed [1]. Identification of smooth surfaces using vision based algorithms has been proposed for singulation of objects [2]. Various pattern recognition and image segmentation techniques have been proposed

for singulation of object in the cluttered environment [3]. Pushing objects at their center of mass in the target direction to isolate the object in the cluttered environment is proposed as an alternative approach for manipulation [4]. King proposed push planners for various tasks related to non-prehensile rearrangements [5]. Physics engine based robotic pushing has been proposed for different scenarios. They utilized object tuning and modelling for the accurate prediction of the behaviors of the object [6]. Reinforcement learning based on the lateral pushing has been used as a technique for singulation of the object in clutter. Robust object grasping in clutter was presented [7]. Studies have been conducted to predict the rigid behavior of the object when manipulated [8–10]. Similarly, self-supervised learning methods have been proposed for grasping objects [11,12]. Few studies used pushing to determine kinematic model of articulated objects and they used the assumption of known segmentation of objects [13,14].

On the other hand, in the dynamic manipulation (i.e., scattering for object singulation), the object keeps moving even after losing the contact with the manipulator. Collision-based scattering is one example of dynamic manipulation to achieve the desired target in short time. Imran et al. [15] proposed a scattering based singulation technique to isolate the objects cluttered in the environment. Using the virtual world simulator based on the dynamic model, they estimated physical properties of all objects in the environment. Based on the desired input from the virtual world, real world scattering could be performed. Approaches related to impulse between robot and the environment have been studied [16,17]. However, dynamic and quasi static manipulations alone cannot achieve the desired object singulation for every possible case. In some cases, combination of dynamic and quasi-static manipulation would be beneficial for effective object singulation. Therefore, this paper proposes a hybrid approach combining dynamic and quasi-static manipulation. This paper is organized as follows. Section 2 explains the methodology of the proposed approach. Modeling and identification of impact are introduced in Section 3. Parameter estimation and dynamics-based simulator are presented in Section 4. Section 5 explains the detail related to dynamic, quasi-static manipulation, and hybrid approach for singulation. Comparison between hybrid approach and pushing only approach is discussed in Section 6. Finally, discussions in 7 and conclusions are drawn in Section 8, respectively.

2. Methodology

In this paper, both non-prehensile and prehensile manipulations will be used to achieve object manipulation in cluttered environment. Soma cube will be used as an example to show the effectiveness of the proposed technique. The Soma cube is a three dimensional puzzle as shown in Figure 1, having seven blocks, made up of 27 cubes. Six blocks are made up of four cubes and one block is made up of three cubes. They are used as the building blocks to assemble them for achieving desired target formation. There are various ways in which these blocks can be assembled such as a cube or several 2D or 3D formations.

Scenario is as follows. Initially, we assume that 7 blocks of the Soma cube are cluttered together in a formation. For grasping blocks safely without colliding with other pieces, singulation of blocks placed on the table should be proceeded first.

Considering pros and cons of the quasi-static singulation and the dynamic singulation, we would like to adapt the following strategy: hybrid approach for efficient object singulation and safe object manipulation.

(i) Step 1: Dynamic singulation approach by scattering
Purpose: separation of objects in short time to make the size of cluster * smaller
(* cluster: a group of blocks that are too close to each other for grasping)
(ii) Step 2: Quasi-static singulation approach by Pushing
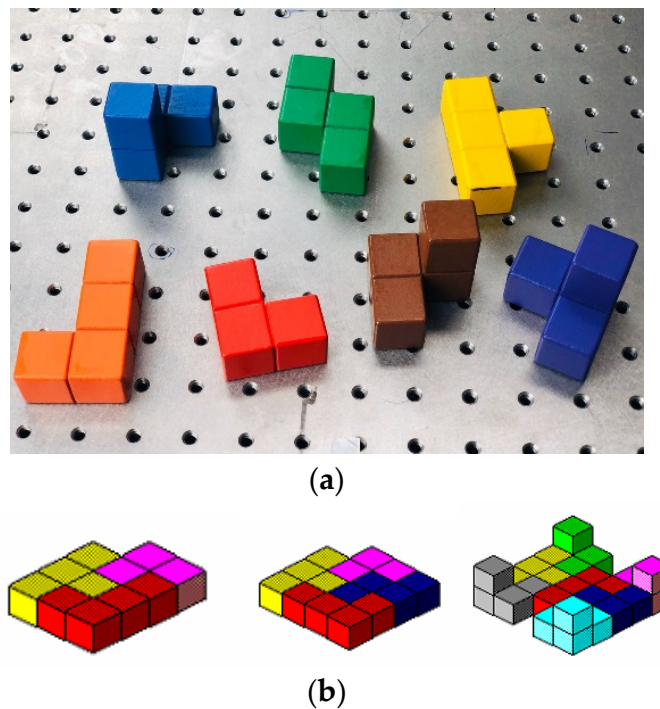Purpose: separation of objects in small cluster group

(a)

(b)

Figure 1. Soma cube. (**a**) Seven blocks (**b**) Different formations.

Figure 2 explains the flowchart of the proposed hybrid singulation approach in detail. Initially, several blocks of Soma cube are laid on the table as shown in Figure 3a. Then, scattering is performed to separate the blocks in a desired manner. Here, the desired manner means scattering all the blocks such that the robotic gripper is able to grasp each block in a safe manner without collision with other blocks remaining on the table. However, just by one time scattering, there is no guarantee that all the blocks can be scattered in a desired manner enough to conduct prehensile manipulation by the robot hand.
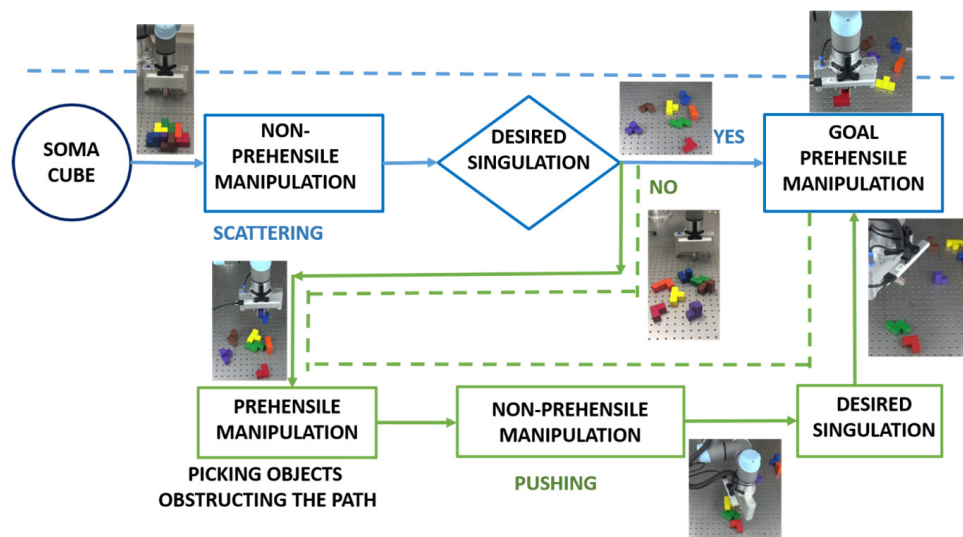

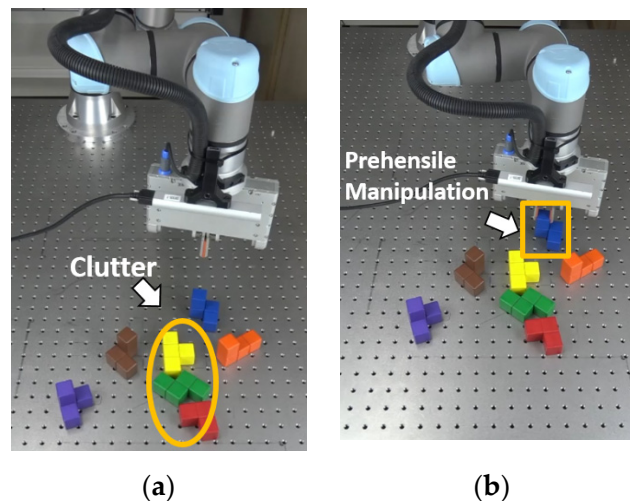**Figure 2.** Hybrid Singulation Manipulation.

Figure 3. (**a**) Clutter (**b**) Prehensile Manipulation.

It is probable that a small cluster group consisting of two or three blocks of Soma puzzle may exist even after scattering. In order to complete the singulation of all blocks, we use pushing as the secondary singulation method. It is, however, noted that before conducting pushing we should perform prehensile manipulation to remove blocks that are obstructing the path of the manipulator when pushing is carried out as shown in Figure 3b. After the desired separation is achieved by pushing, prehensile manipulation is conducted for the remaining pieces. As explained in the proposed approach, the prehensile and non-prehensile manipulation methods are combined for effective robot grasping and manipulation of blocks in cluttered environment. Furthermore, in order to implement non-prehensile manipulation such as dynamic and quasi-static methods, we need to identify the dynamic characteristics of blocks and the environment.

Based on these properties, a virtual world simulator is designed to perform the object singulation. The virtual world simulator is designed based on the proposed impulse based dynamic model. In the virtual world, first of all the point of contact (i.e., where to hit), the piece to hit, and the direction of velocity are decided. Under this condition, the velocity of manipulator is controlled based on the feedback error (i.e., the distance error between the pieces) until the desired object singulation is achieved. In the virtual world simulator, the colliding velocity of robot and direction of collision is updated by employing a feedback routine to minimize the error to achieve the desired singulation pattern. The physical properties of the blocks such as masses, coefficient of restitutions between blocks, and coefficient of friction between blocks and the ground are estimated based on the technique proposed by Imran et al. [15].

If the blocks are still in contact after scattering, then the pushing method is applied. In the virtual world, the velocity is increased gradually until the scattering creates required separation between blocks. However, even though the velocity is increased to a maximum level, there are some cases for them, the required separation cannot be achieved. In case of scattering, we have used virtual world simulator and in case of pushing we have used the image segmentation based principal component analysis (PCA) approach. For few blocks still in contact with each other, quasi-static manipulation based on PCA algorithm is performed to complete the object singulation.

3. Modeling, Identification and Condition for Dynamic and Quasi-Static Manipulation

This section introduces impulse modeling first. Using the impulse model, we distinguish the behavior of blocks and robot manipulators just after collision in terms of scattering and pushing.

3.1. Impulse Modeling

An analytical impulse based approach is considered which is based on the dynamic manipulation of the blocks. We assume that the robot manipulator collides with one of the blocks being cluttered in the environment. Then the impulse is propagated from one block to other blocks sequentially. For collision between the manipulator and block or collision among different blocks, the corresponding impulse based model has been developed in [15] and we have summarized the derivation in the following for the convenience of the reader.

Impulse propagation can be seen in Figure 4, where manipulator scatters the blocks. In case of known coefficient of restitution, the following equation can be used to calculate the incremental change in relative linear velocities of two colliding bodies

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

where e is the coefficient of restitution. v_I and v_{env} are the absolute linear velocities of the robot and the environment (here, it will a block), respectively. Similarly, Δv_I and Δv_{env} denotes the velocity increments of manipulator and the environment after the impact and T represents the transpose of vectors and matrices, respectively. If we consider a case where there is no friction between the contacting surfaces, the impulse will always act along the normal vector **n** at the contact point. The inverse dynamic model of the robot manipulator with respect to independent joint set is given as follows

$$\tau_a = [I_{aa}]\ddot{\varphi}_a + \dot{\varphi}_a^T[P_{aaa}]\dot{\varphi}_a + g_a - \left[G_a^I\right]^T F_{ext} \tag{2}$$

where $[I_{aa}]$ and $[P_{aaa}]$ denotes the inertia matrix and inertial power array with respect to the independent joint set $\dot{\varphi}_a$, respectively. $\left[G_a^I\right]$ and $[G_a^{env}]$ represents the Jacobian matrices of robot manipulator and environment, and *env is* the abbreviation of environment, respectively. τ_a, F_{ext} and g_a represent the joint torque vector, the effective force vector, and the gravity load, and a represents actuated joints in robot manipulator, respectively. Position and velocities are finite during the impact, consequently the term $\dot{\varphi}_a^T[P_{aaa}]\dot{\varphi}_a$ will be zero at $\Delta t \rightarrow 0$. Then, the incremental joint velocity vector can be expressed as

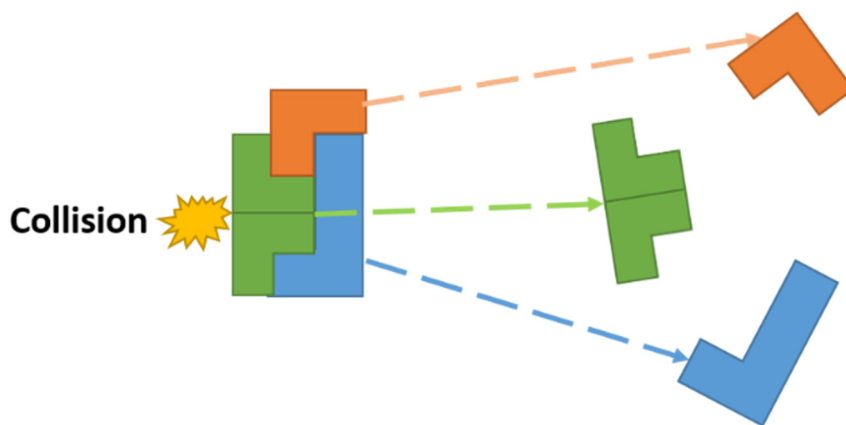$$\Delta \dot{\varphi}_a = [I_{aa}]^{-1}\left[G_a^I\right]^T \hat{F}_{ext} \tag{3}$$

Figure 4. Illustration of collision between robot and objects during singulation.

Using the forward kinematic relation given by $\Delta v_I = \left[G_a^I\right]\Delta\dot{\varphi}_a$, the contact point velocity vector of the robot manipulator is given as follows

$$[\Delta v_I] = \left[G_a^I\right][I_{aa}]^{-1}\left[G_a^I\right]^T\left(\hat{F}_{ext}\right), \tag{4}$$

Similarly, the contact point velocity incremental for environment can be represented as

$$[\Delta v_{env}] = [G_a^{env}][I_{env}]^{-1}[G_a^{env}]^T\left(-\hat{F}_{ext}\right) \tag{5}$$

Substituting (4) and (5) into (1), we have the final closed-form solution of the external impulse as follows.

$$\hat{F}_{ext} = \frac{-(1+e)(v_I - v_{env})^T n}{n^T\left\{\left[G_a^I\right][I_{aa}]^{-1}\left[G_a^I\right]^T + [G_a^{env}][I_{env}]^{-1}[G_a^{env}]^T\right\}n}, \tag{6}$$

where the first and second terms in the denominator are associated with the dynamics of the manipulator and environment, respectively.

When the robot manipulator will collide with the block A, then the block A will collide with other block (like the block B) in the environment. Here, we assume the central impact. That is, the friction between two colliding bodies is negligible. Consider two bodies with mass center velocities v_a, v_b and angular velocities ω_a, ω_b are approaching and contacting to each other. Then the dynamic model of each block can be described as

$$F_{ext} = m_a\dot{v}_a \xrightarrow{\int \cdot dt} \hat{F}_{ext} = m_a(v'_a - v_a) \tag{7}$$

$$-F_{ext} = m_b\dot{v}_b \xrightarrow{\int \cdot dt} -\hat{F}_{ext} = m_b(v'_b - v_b) \tag{8}$$

and

$$r_a \times F_{ext} = I_a\dot{\omega}_a \xrightarrow{\int \cdot dt} r_a \times \hat{F}_{ext} = I_a(\omega'_a - \omega_a) \tag{9}$$

$$-r_b \times F_{ext} = I_b\dot{\omega}_b \xrightarrow{\int \cdot dt} -r_b \times \hat{F}_{ext} = I_b(\omega'_b - \omega_b) \tag{10}$$

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.

Then, the contact point velocity vectors of both bodies are written as follows

$$\overline{v}_a = v_a + \omega_a \times r_a, \tag{11}$$

and

$$\overline{v}_b = v_b + \omega_b \times r_b, \tag{12}$$

where r_a and r_b denote the vectors directing from the center of the objects to the point of contact p.

Finally, Equations (7) to (10) are rearranged to find the velocity increments of both bodies in terms of external impulse as follows

$$\Delta v_a = \hat{F}_{ext}m_a^{-1} + (r_a \times \hat{F}_{ext})I_a^{-1} \times r_a, \tag{13}$$

and

$$\Delta v_b = \hat{F}_{ext}m_b^{-1} - (r_b \times \hat{F}_{ext})I_b^{-1} \times r_b, \tag{14}$$

where $\Delta v_a = \overline{v}'_a - \overline{v}_a$ and $\Delta v_b = \overline{v}'_b - \overline{v}_b$. $\overline{v}'_a$ and $\overline{v}'_b$ denotes the post impact velocity vectors at contact point of body 'A' and body 'B', respectively. Similarly, to (6), the closed-form solution of the external impulse is obtained as follows:

$$\hat{F}_{ext} = \frac{-(1+e)(\overline{v}_a - \overline{v}_b) \times n}{m_a^{-1} + m_b^{-1} + [(r_a \times n)I_a^{-1}) \times r_a + (r_b \times n)I_b^{-1}) \times r_b] \times n}. \tag{15}$$

3.2. Conditions for Quasi-Static and Dynamic Manipulation

In this section, we will analyze the conditions for both quasi-static and dynamic manipulations. The robot manipulator UR5 and Soma puzzle cube are involved for the implementation of the impulse based model. In this case, the inertia of the robot manipulator is always much greater than the mass of the soma puzzle cube. It implies that the change in the end-effector velocity of the manipulator is much smaller as compared to the change in the velocity of the soma puzzle cube. Therefore, the external impulse given in (6) can be simply expressed as

$$\hat{F}_{ext} = -(1+e)m_a(v_I - v_a). \tag{16}$$

The dynamic equation of the block A is described as

$$\sum F = \hat{F}_{ext} - \mu m_a g = m_a \dot{v}_a \tag{17}$$

where F_{ext} and $\mu m_a g$ denotes the external impulsive force exerted on the block A by the robot and the friction force between the block and the ground. Integrating (17) in very short period of time, it becomes

$$\hat{F}_{ext} = m_a \dot{v}_a - \mu m_a g \xrightarrow{\int \cdot dt} \Delta v_a = m_a^{-1} \hat{F}_{ext} \tag{18}$$

Substituting (16) into (18) yields

$$\Delta v_a = (1+e)(v_I - v_a). \tag{19}$$

Since $\Delta v_a = v_a' - v_a$, the general form of the block A's velocity vector after collision can be written as follows.

$$v_a'(t_o) = -ev_a(t_o) + (1+e)v_I. \tag{20}$$

where $v_a(t_o)$ represents the initial velocity of the block A which is zero in this case at time t_o. $v_a'(t_o)$ is the velocity of the block A just after collision with the robot manipulator.

Since the motion of the block A just after impact is governed by (17), the velocity of the block A after collision with the manipulator can be described as

$$v_a'(t) = v_a'(t_o) - \frac{v_a'(t)}{|v_a'(t)|} \mu_a g t + 1/m_a \int_{t_o}^{t} F_{ext} dt \tag{21}$$

by integrating (17) in time. Here, F_{ext} becomes zero when the robot loses contact with the block A.

As a result of collision, the robot and the block A can be separated or still stick each other. We can interpret this behavior analytically.

It is noted that pushing and scattering are associated with the relative velocities of the robot and blocks after collision. If the robot and the block A stick together, it can be considered as pushing. Then, the following equation will hold the robot pushes the block A,

$$v_I' = v_a'(t_o), \tag{22}$$

where v_I' is the velocity vector of the end effector of robot just after collision.

The above equality implies that the contact point velocity of the robot is equal to the velocity of the Soma block.

In case of scattering, the following equation will hold the block A moves faster than the robot

$$v_I' < v_a'(t_o), \tag{23}$$

To know the final position of the block A in case of scattering, we need to know the dynamic parameters of the block A, coefficient of friction and coefficient of restitution. On the other hand, in case of pushing we only need to know the motion of manipulator to know the motion of the block A and this motion is independent of the dynamics of the block A.

4. Parameter Estimation and Simulator Based Dynamic Manipulation

4.1. Estimation of Physicial Parameters

The complete methodology to estimate the physical properties of objects such as coefficient of friction, coefficient of restitution, and mass ratios is proposed by Imran et al. [15]. Here, multiple videos of different impact scenarios were recorded by considering the collision of single and two objects with robot manipulator. By processing these unlabeled videos, certain parameters such as travel distance of each object after collision, travel time, velocities before/after collision, and acceleration of each object were observed. Furthermore, using these observed motion parameters, and combining with the analytical model, the physical properties of all the objects being involved in the collision could be estimated.

Based on the same methodology, we have estimated the physical properties of the objects for our Soma cube application. These physical properties have been used to design virtual world simulator and to perform experimental singulation. In the experimentation, 7 Soma cube blocks are used. Universal robot UR5 manipulator is used for scattering and pushing. A 4K FDR-AX30 (Sony, Tokyo, Japan) is used for recording videos for the estimation of physical parameters.

4.2. Coefficient of Restitution

In order to calculate the coefficient of restitutions (COR) between object and manipulator and among objects, we have recorded several videos. To find COR between objects and manipulator, several videos are recorded by colliding each object with robot manipulator. In this case, the colliding velocity of the manipulator is known, and the velocity of the object is calculated by processing the recorded videos. the coefficient of restitutions between each object and robot manipulator is calculated using following equation

$$v_a{}' - v_a = -\hat{F}_{ext}/m_a = -(1+e)v_I, \tag{24}$$

where v_a and v_a' are the velocities of object before and after collision, respectively. v_I is the velocity of the robot manipulator which is known, and $v_a = 0$ because the object is initially at rest and v_a' is calculated from videos for each object. In order to calculate the coefficient of restitution among objects the following equation is used.

$$e = \frac{v_a{}' - v_b{}'}{v_b - v_a}, \tag{25}$$

where v_a and v_b are the velocities of objects "a" and "b" before collision and v_a' and v_b' are the velocities of objects "a" and "b" after collision.

In this case, the videos are recorded by making the collision of two objects with robot manipulators. The velocity of the objects before and after collision is calculated by processing the record videos.

Figure 5 shows the experimental results: coefficient of friction µ (o-T) between soma block and the table surface, the coefficient of restitution between objects e (o-o), and the coefficient of friction between object and manipulator e(o-m), respectively. The mass ratio among 7 objects of the soma puzzle was also estimated and its value ranges from 0.95 and 1.05.
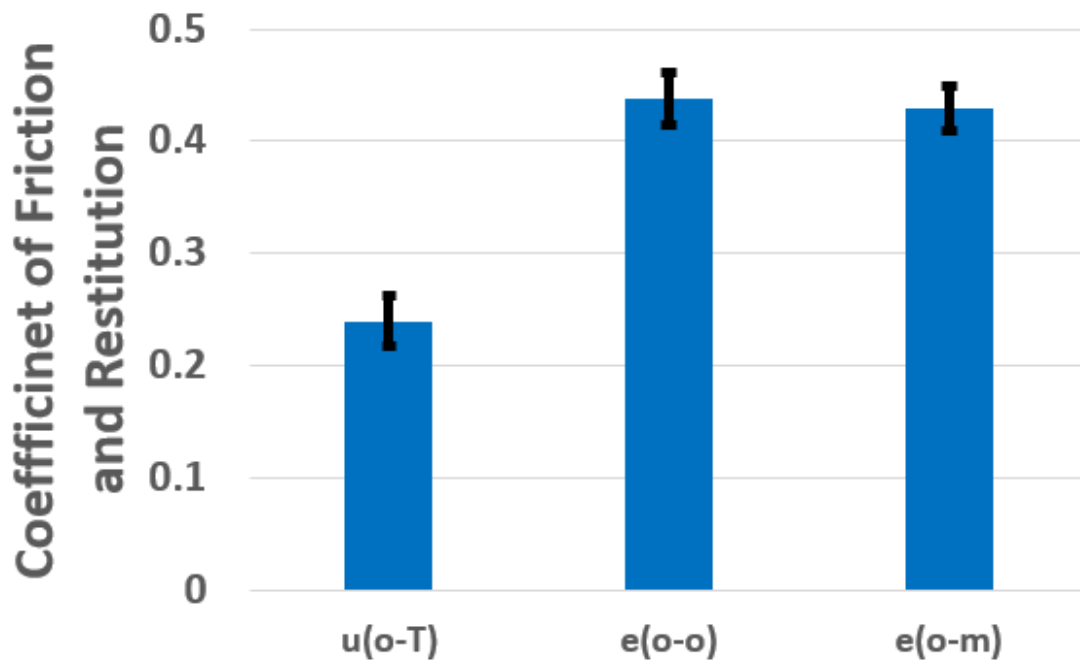
Figure 5. Coefficients of friction and restitution.

4.3. Simulator-Based Dynamic Manipulation

Control singulation can be performed, without disturbing the real-world environment, by using a virtual world simulator. Since singulation is an irreversible process, so iterative singulations for the same scenario is not possible in real world experiment. Contrarily, virtual world simulator would allow us to perform iterative singulations for the same scenario. In order to perform the control iterative singulation in virtual world simulator, the initial configurations of pile of objects would be scanned from real world environment by using camera and initial positions and physical properties' information of each object will be given in virtual world simulator. Finally, iterative control singulation would be performed in virtual world. Once the desired singulation is achieved in the virtual world at specific colliding velocity, the same velocity would be given to the robot manipulator in real world to perform the singulation

In this paper, virtual world simulations have been done using a physics engine: a commercially available multi-body simulation software called "Daful", which is made by Virtual-motion Co. [18]. These virtual world simulations have been performed by integrating the estimated physical properties such as mass ratios, coefficient of restitution and coefficient of frictions to perform the control singulation of objects. Control singulation implies to achieve the desired minimum distance among objects. The objective is to achieve the minimum distance among objects so that it is enough to grasping them by gripper. The colliding velocity of robot and direction of collision is updated in virtual world by employing a feedback routine that would minimize the error to achieve the desired singulation pattern [15]. The error is calculated by comparing the minimum distance of objects in each iteration in virtual world with the minimum desired distance among objects. Once, the minimum distance among object is achieved in virtual world, the same velocities information is used in real world to perform the object singulation. However, for large pile of objects such as for case of seven objects, it is not always possible to achieve the desired minimum distance among objects within predefined workspace. In those cases, the velocity of the manipulator is continuously increased, until the blocks leave the predefined workspace. That maximum threshold velocity is applied in real world to perform the singulation. The blocks, which remained in contact even after scattering, are singulated by using the pushing technique without using the virtual world simulator.

5. Dynamic, Quasi-Static Manipulation, and Hybrid Approach for Singulation

We will test three cases.

(i) Case 1: complete singulation of blocks by one time scattering
(ii) Case 2: complete singulation of blocks by pure pushing
(iii) Case 3: incomplete singulation of blocks by one time scattering, followed by several steps of pushing

5.1. Dynamic Manipulation or Singulation by Scattering

To validate the proposed methodology, we have increased the difficulty level from Figures 6–9. In Figure 6 three blocks are used, in Figure 7 four blocks are used and in Figures 8 and 9 seven blocks are used. The successful singulations are performed in the virtual world based on proposed algorithm and then the same singulation is achieved in real world by using the magnitude and direction of velocity for manipulator from the virtual world. However, for the case of seven objects some time it is not possible to singulate using scattering alone as shown in Figure 9. That is where the hybrid approach comes in where pushing is performed followed by the scattering to achieve the desired singulation.
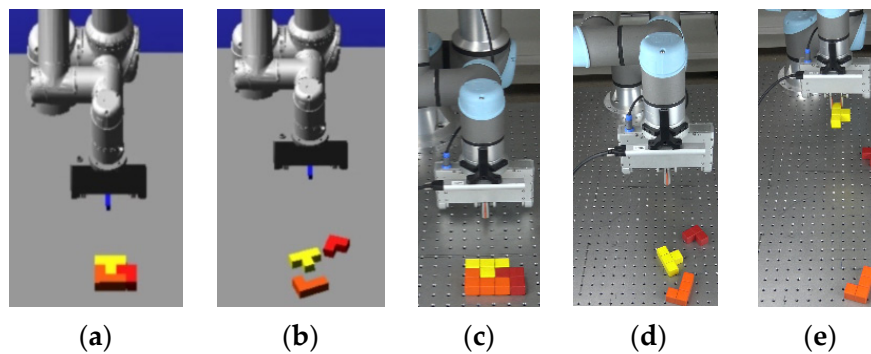

Figure 6. Soma cube: three blocks formation. (**a**) Virtual world Soma blocks: initial position, (**b**) Virtual world Soma blocks: final position after scattering, (**c**) Real world Soma blocks: initial position, (**d**) Real world Soma blocks: final position after scattering, (**e**) Prehensile manipulation after singulation of Soma blocks.
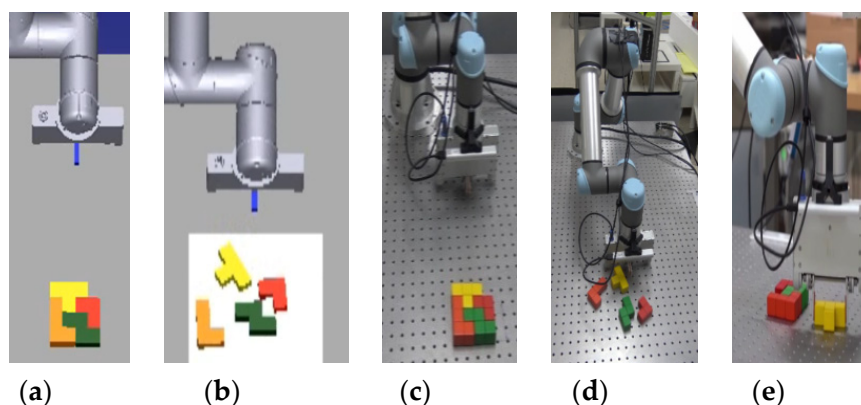

Figure 7. Soma cube: four blocks formation. (**a**) Virtual world Soma blocks initial position, (**b**) Virtual world Soma blocks final position after scattering, (**c**) Real world Soma blocks initial position, (**d**) Real world Soma blocks final position, (**e**) Prehensile manipulation after singulation of Soma blocks.
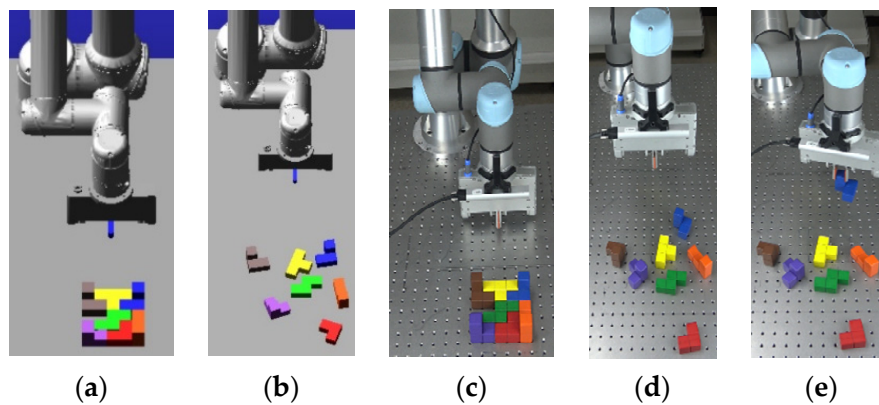
(a) **(b)** **(c)** **(d)** **(e)**

Figure 8. Soma cube: seven blocks formation. (**a**) Virtual world Soma blocks: initial position, (**b**) Virtual world Soma blocks: final position after scattering, (**c**) Real world Soma blocks: initial position, (**d**) Real world Soma blocks: final position after scattering, (**e**) Prehensile manipulation after singulation of Soma blocks.
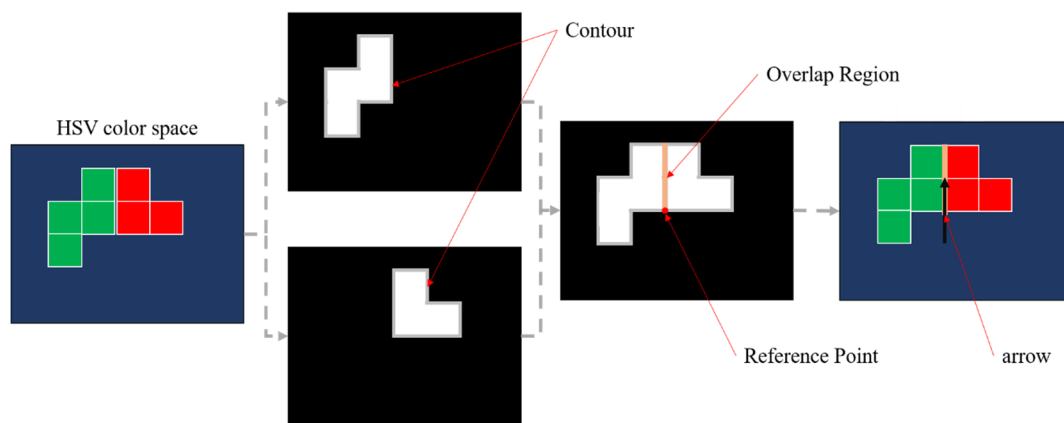

Figure 9. Principal component analysis (PCA) based image segmentation.

In case 1, we employ three, four, and seven blocks formation. What will be done in the experiment is that the robot scatters all blocks by hitting the initial formation. There are three parameters when conducting scattering: the position of the collision, the direction of collision, and the velocity, which are being updated by feedback routine. We can conduct scattering in the virtual world same as the real world since we already identified all parameters of blocks and environment. Real world orientation of the soma block formation will be fed to virtual world using the camera attached with manipulator and scattering is performed in virtual world until it will create separation among blocks. These parameters form virtual world are then fed to real world to perform singulation. Figures 6–8 demonstrate the successful singulation by just one time scattering and prehensile manipulation to move the blocks to other places.

5.2. Quasi-Static Manipulation or Singulation by Pushing

In case 2, we employ seven blocks. It is found that singulation of all seven blocks takes time in pushing only singulation.

5.2.1. Image Segmentation

In this paper, we have used the quasi static manipulation which is based on the image processing techniques. RealSense SR305 Depth Camera (Intel, California, USA) is used to take image and it provides highly accurate depth data in a wide variety of vision applications. First, we take an RGB image from the camera and converts it into an Hue, Saturation, Value (HSV) color space. When expressed

in the HSV color space, the brightness of the image may be less affected than the RGB color space. Thus, it allows effective judgement of the specific color. Next, the cubes were segmented according to the HSV range. For example, if the cube is in the red color, the area is set to 255, and in all other cases, it is set to 0. Since the Soma Cube blocks are of single color, segmentation is relatively easy on complex backgrounds. Accordingly, the results of segmentation for each color (i.e., for each piece) are stored for each image. The saved images are post-processed by closing technique to remove noise. The closing technique is to perform erosion immediately after dilation and return to the original image size, and the contour of the image is obtained as a result of each segmentation. Here, the image contour is a line connecting the edge borders of parts having the same color or the same intensity. Furthermore, the thickness of the contour is made thick so that the contours can overlap if there are adjacent cubes. Now we compare the images of each piece in pairs to find intersection (or overlapping) between two contours of two pieces. However, there are noises even in overlapped images. When the contoured area was smaller than a certain area threshold, additional processing is required, and the bottom point of the overlapped contour part was taken as the reference point to mark the arrow after applying principal component analysis (PCA) to the region of overlapping contour. Here, PCA is a method of finding the principal component of the distributed data and is used for reducing the dimensionality of datasets, increasing interpretability but at the same time minimizing information loss and the bottom point refers to the point with the highest value of the row of the image. 2D image coordinate consists of row and column and the origin is located at the upper left corner. In our system, the camera mounted manipulator takes a picture and then goes back to home position where it will follow the direction of the arrow to perform pushing. In our two-dimensional binary image, we obtain two vectors that best represent the "255" intensity region, where 255 is changed to 1 and refers to white color, and 0 refers to black color. The two vectors represent the longest axis with the largest variance of the region and the shorter axis with the second largest, among which we chose the longest axis as the direction of pushing vector. The end point of the overlapped contour part in the row direction was used as the center point. Moreover, considering the size of the gripper, the length of vector pointing the direction of pushing was increased by the length of end effector. So, we changed the start point and end point of the arrow to 3D coordinates of the camera point of view, respectively, and pushed it out using the gripper.

5.2.2. Pushing Algorithm

Pushing algorithm is proposed as Figure 10. In the "Select Target" stage, we interpret the scene of camera. The image taken from RealSense Depth Camera SR305 (Intel, California, USA) is divided into several groups; group with several blocks in clutter and group with isolated blocks as shown in Figure 9. In "Select Action" stage, we figure out which approach to be used. If some blocks are contacting each other, then we will use the non-prehensile manipulation such as pushing to singulate the blocks. Lastly, we remove the isolated Soma blocks one by one starting from the block nearest to the robot position.

In pushing only singulation, the pushing will be performed with prehensile manipulation until all the blocks are completely singulated. When there are many possible pushings (as will be the case, for example, in the initial formation, and also later steps), we choose the overlapping contour region nearest to the end effector. After each pushing, the robot manipulator takes the image using real sense and process it to perform the next pushing. In Figure 11, the overlapping contour region between yellow and blue blocks is closest to the end effector. So, the first pushing will be along the green arrow between these two blocks as shown in Figure 11b. After each pushing, the resulting formation is analyzed as discussed before, and then the pushing is repeated just described, until all blocks are completely isolated.
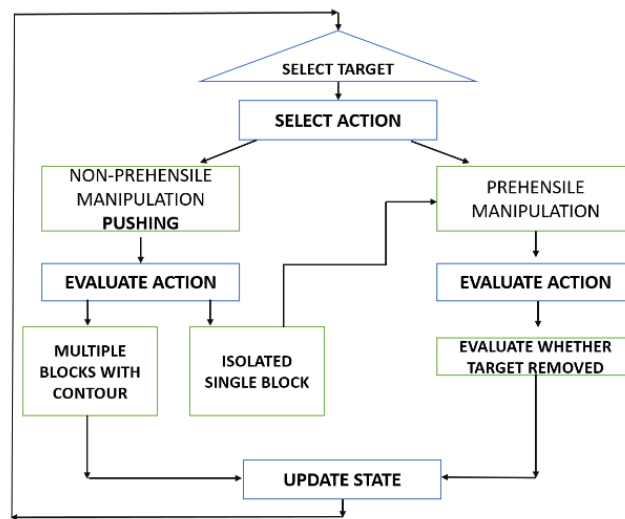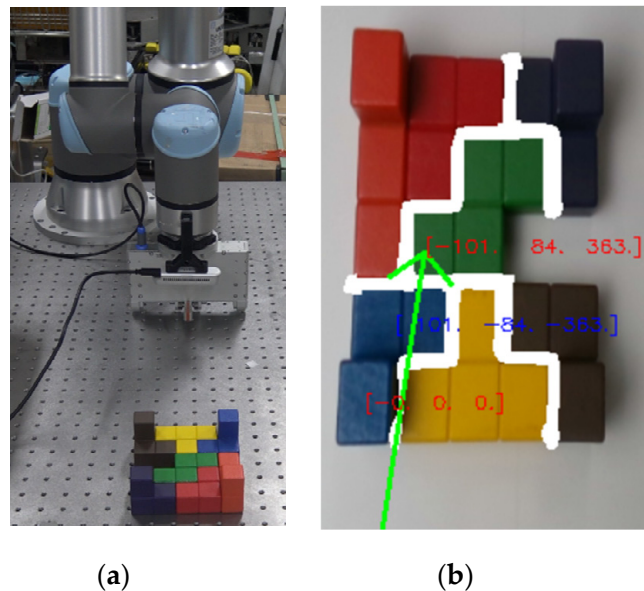
Figure 10. Quasi-static manipulation.

(a)   **(b)**

Figure 11. Pushing (**a**) Manipulator and Soma blocks in clutter (**b**) Direction of pushing in quasi-static manipulation.

5.3. Hybrid Singulation

In case 3, we employ seven blocks. It is found that singulation of all seven blocks is hard to achieve always. Initially, the real world formation is fed to the virtual world and the direction of scattering is decided on the basis of robot's ability to generate maximum impulse in a certain direction. Along this direction, the colliding velocity of robot is updated in the virtual world by employing a feedback routine that would minimize the error to achieve the desired singulation pattern. After achieving the singulation pattern in virtual world, the direction and velocity are applied in real world to achieve desired singulation. However, in most cases, only a partial scattering can be achieved. The advantage of the initial scattering is that we can scatter the initial cluster of seven blocks into several small groups.

Figure 12, shows the hybrid manipulation, after initial scattering the blocks are not completely singulated so we have used quasi-static manipulation in which Intel RealSense Depth Camera was used that has created contours among the blocks which are very near or their boundary touching each other. As shown in Figure 12c, two contours appear in this case, one between yellow and green block

and the second contour between green and red block. All the other blocks are isolated after scattering. The robot manipulator picks each isolated block having no overlapping contour starting from the block nearest to the end gripper.
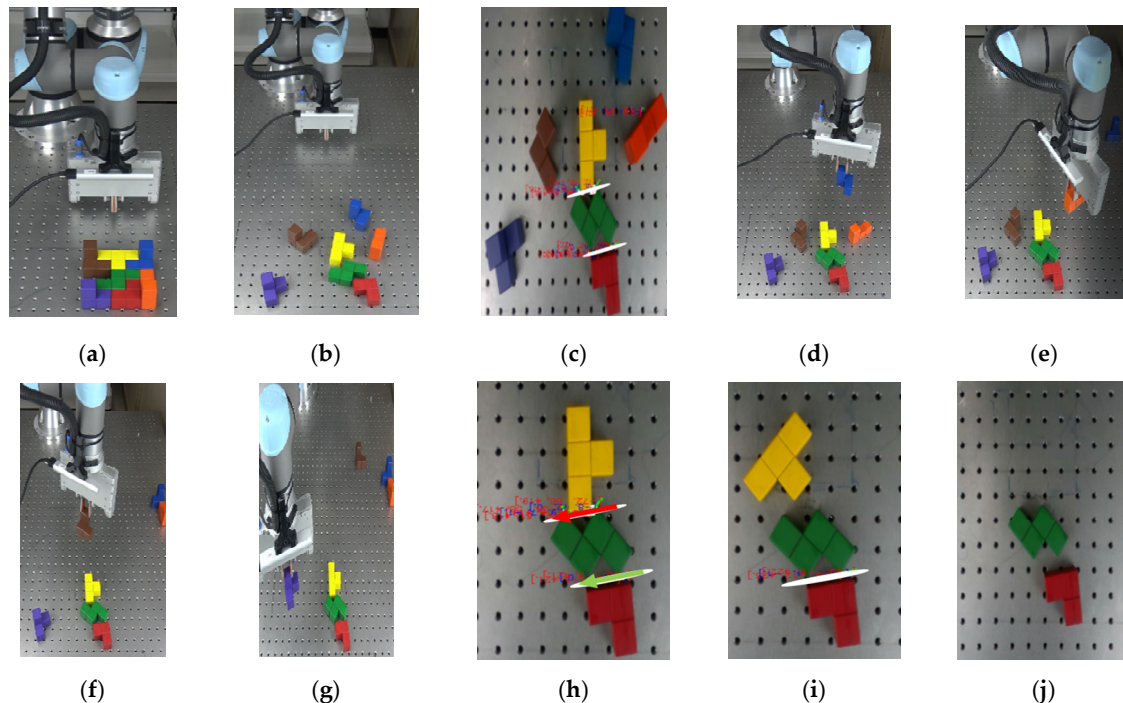

| (a) | (b) | (c) | (d) | (e) |

| (f) | (g) | (h) | (i) | (j) |

Figure 12. Hybrid Singulation (**a**) Real world soma blocks: initial position (**b**) Real world soma blocks after scattering (**c**) Two contours shown in white color among red green and yellow blocks (RealSense View). (**d**) Prehensile manipulation for clearing the isolated blue block (**e**) Prehensile manipulation for clearing the isolated orange block (**f**) Prehensile manipulation for clearing the isolated brown block (**g**) Prehensile manipulation for clearing the isolated purple block (**h**) Three blocks with two overlapping contours and arrows showing the direction of pushing (**i**) After one pushing, yellow block separated (**j**) After second pushing, both green and red block get separated.

After removing all the isolated blocks, the manipulator performs pushing as shown by the arrows above the contours in Figure 12h. The manipulator will follow the direction of the vector created by using image segmentation technique that is closer to manipulator. In this case, the yellow block gets isolated first. After first pushing, RealSense Depth Camera SR305 (Intel, California, USA) again captures the image and robot manipulator will decide whether the blocks get isolated or still touching each other. In the second push, the green and red blocks are separated as shown in Figure 12j.

The Figure 13 shows another example of the hybrid singulation, where the blocks were scattered after dynamic manipulation. All the blocks are scattered except green and red blocks and they are removed using prehensile manipulation, leaving only green and red blocks. Using the PCA based image segmentation algorithm, the direction of pushing is easily generated which is shown by green arrow in which the tail of the arrow shows the starting point and head shows the ending point of the pushing. The manipulator initially removes the isolated blocks nearest to the manipulator using the proposed quasi-static manipulation algorithm as explained in Figure 10. Accordingly, the blocks are removed by the order of blue, orange, yellow, brown, and purple. Now the manipulator will use pushing technique to isolate the red and green blocks that has formed a cluster of two blocks.
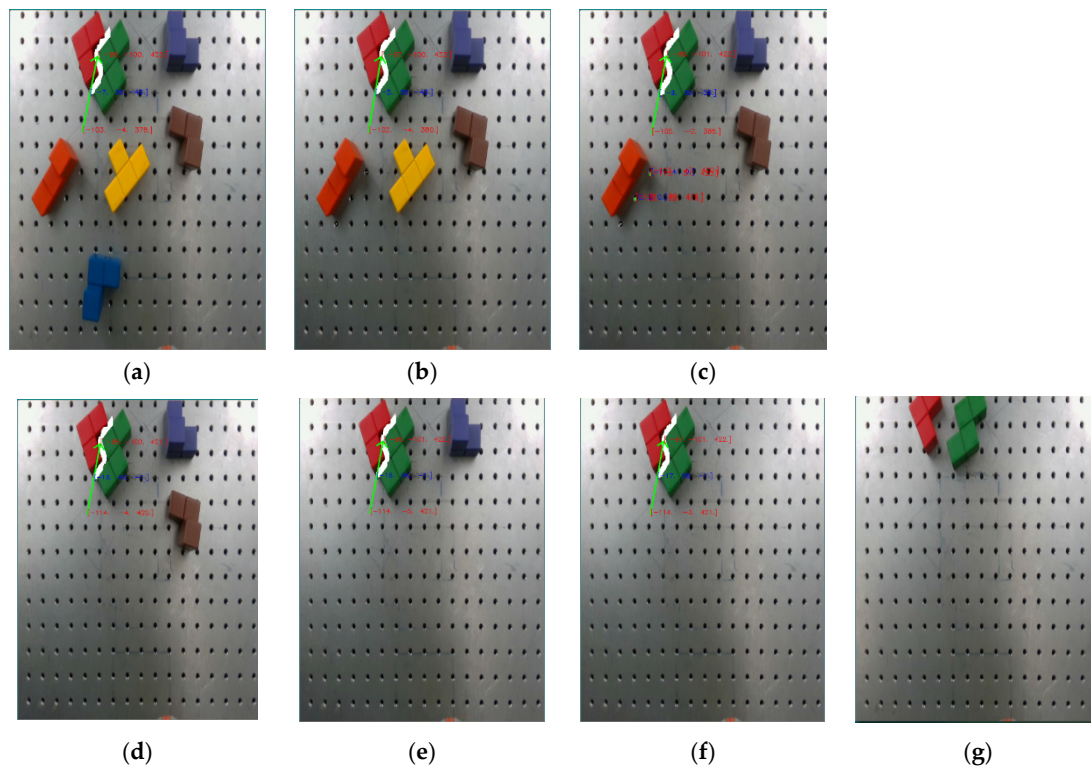
Figure 13. Hybrid Singulation. (**a**) After initial scattering, two blocks in a cluttered group and other isolated blocks are shown. (**b**)–(**e**) Each of the separated blocks are removed by prehensible manipulation. (**f**) Purple block is removed, and pushing direction between red and green blocks is shown. (**g**) After pushing, red and green blocks are separated.

6. Comparative Analysis between Hybrid Singulations vs. Pure Pushing and One Time Scattering Singulation

In this section, we have provided the comparative analysis among the approaches: hybrid singulation, pure pushing and one time scattering based singulation. In order to validate the effectiveness of the hybrid singulation, we conducted experimentation with three, four, and full seven block formations; seven experiments were conducted for each case. For three and four blocks formation, one time scattering can achieve singulation and it takes 0.5 s to perform one time scattering. With the increase in the number of blocks, one time scattering approach cannot singulate Soma blocks for all different formations. In case of seven blocks formation for seven different formations only one time we achieve successful singulation out of seven experiments using one time scattering approach. Pure pushing approach can singulate blocks however, it is time consuming. Hybrid approach is the most effective since it utilizes both pushing and scattering together and achieve singulation in less time as shown in Table 1. We have measured the time for all cases. In case of hybrid singulation for seven blocks, the average task completion time was 6.7 s and in case of pure pushing it was 30.2 s. The standard deviation and coefficient of variation for pure pushing were 2.69 and 8.88, respectively. The standard deviation and coefficient of variation for hybrid approach were 2.13 and 31.84 respectively. It is observed that hybrid approach has less standard deviation but that coefficient of variation is larger. That implies that pure pushing is more consistent in terms of task completion time, though it took more time for singulation of seven objects.

Table 1. Comparison between pure pushing and hybrid singulation.

Formations	7 Block (Pure Pushing)	7 Block Hybrid Approach
Number of Experiment	5	5
Success/Fail	5/0	5/0
Average Time (sec)	30.28	6.71
Standard Deviation	2.69	2.13
Coefficient of Variation (CV)%	8.88	31.84

In Figure 14a it took manipulator 10 pushing to completely singulate all the blocks, similarly in Figure 14b,c it took seven pushing to completely singulate all the blocks. In case of Figure 15 we have used same formations as used in pure pushing cases and tried to singulate them using hybrid approach and in all the three cases it took single pushing after initial single scattering to singulate all the blocks. Hybrid singulation approach comprises of one scattering and one pushing on average. In case of pure pushing, it takes on average seven pushing to perform complete singulation. That is why pure pushing technique takes more time.
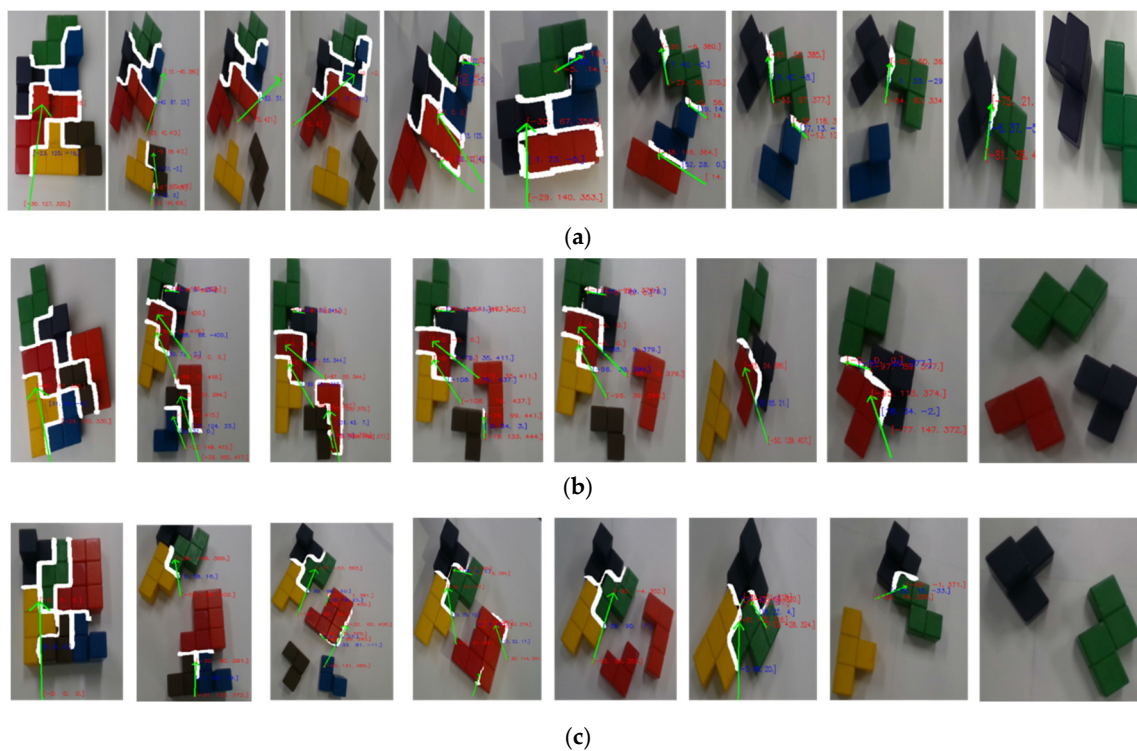

(a)

(b)

(c)

Figure 14. Pure Pushing with three different formations and green arrow showing direction of pushing. (**a**) Initial formation for first case to final singulation takes 10 pushing. (**b**) Initial formation for the second case to final singulation takes 7 pushing. (**c**) Initial formation for the third case to final singulation takes 7 pushing.
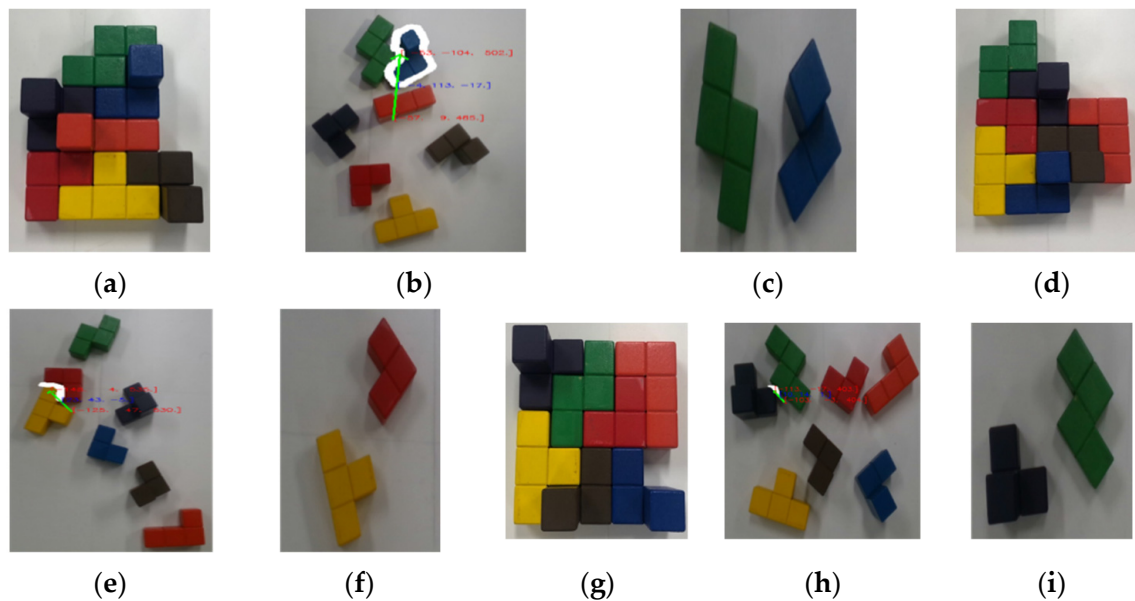
Figure 15. Hybrid Singulation with three different formations. (**a**) Initial formation of first case of 7 soma blocks. (**b**) Formation after scattering showing the need of single pushing and green arrow points in the direction of pushing between green and blue blocks, (**c**) green and blue blocks are singulated after single pushing, (**d**) Initial formation of second case of 7 soma blocks, (**e**) Formation after scattering showing the need of single pushing and green arrow points in the direction of pushing between yellow and red blocks, (**f**) yellow and red blocks are singulated after single pushing, (**g**) Initial formation of third case of 7 soma blocks, (**h**) Formation after scattering showing the need of single pushing and green arrow points in the direction of pushing between green and purple blocks, (**i**) green and purple blocks are singulated after single pushing.

7. Discussion

In this paper, we have proposed the hybrid singulation combining scattering and pushing. This framework uses both prehensile as well as non-prehensile manipulation to achieve desired singulation. In the initial stage, we performed scattering based manipulation to achieve singulation of blocks in clutter. After scattering, all blocks may not be singulated. So, in the second stage, we conduct pushing. Prior to pushing, we need to remove the blocks that may collide with the blocks in clutter and hinder pushing. So, we have use prehensile manipulation to pick the isolated soma blocks. Then, we perform pushing to achieve the desired object singulation.

Mehmet et al., provided a framework for non-prehensile manipulation under clutter [19]. In this approach, they have used pushing as a non-prehensile manipulation method to manipulate objects in clutter. In this approach, they had termed some spaces as invisible or unknown because the clutter had blocked the sensors (cameras or laser scanners). However, in our approach we do not face any such spaces and our sensors are not blocked and after each push and scattering, our manipulator having RealSense Depth Camera SR305 (Intel, California, USA) takes picture and move the manipulator based on the pushing directing of the arrow generated by image segmentation technique.

A framework of push-grasping in clutter is an approach for planning in clutter that is based on human strategies [1]. However, in their approach they have experienced the problem of unnecessary displacement of objects. The hybrid singulation approach proposed in our paper allows the blocks to remain in the workspace because if we rely only on scattering, we can also face the similar problem, but the hybrid approach solve the problem by using pushing once the scattering has been done and all the block remain in the workspace and there is no unnecessary displacement of objects that can create hindrance for other blocks.

Several factors affect the time required for singulation and these are: the direction for robot scattering, appropriate light for image processing algorithm, orientation of objects, and number of objects. The direction for robot scattering was selected on the basis of the ability of the manipulator to generate impulse in certain direction. The direction of robot scattering was decided to apply the maximum impulse [15]. Appropriate light in the environment allows the image processing algorithm to adequately point out the direction of pushing. For three or four blocks formation, if the orientation/position of the blocks relative to the robot changes, the time to execute the scattering will not change too much as long as the robot can hit the blocks at the chosen point along the chosen direction. When the number of blocks are more than four, one time scattering approach cannot acheive desired singulation for all various different formations of Soma blocks. The pushing can achieve desired singulation but it is time consuming. Our proposed hybrid approach not only achieve desired singulation but also conumes less time as shown in Table 1.

In the current approach, our emphasis is on acquiring desired singulation among soma blocks so that every block can be easily picked using prehensile manipulation. In the future, we would like to extend this study to target based hybrid singulation: where a desired target object can be singulated using the current approach.

8. Conclusions

In this paper, both prehensile and non-prehensile manipulations are used for object singulation in cluttered environment. Non-prehensile manipulation is divided into quasi-static and dynamic manipulation. We proposed a hybrid singulation approach combining the merits of both dynamic scattering and pushing approach. The Soma cube was employed to corroborate the effectiveness of the proposed algorithm. The proposed hybrid singulation approach achieves desired singulation in less time when compared to pure pushing approach. It is expected that the proposed hybrid object singulation algorithm could be applied to general application areas. For the future work, we will consider the following, First, to assemble the Soma cube in a given formulation, the blocks need to be removed and assembled in a specific, not a random order, which is not considered in the paper. Moreover, the collision with other obstacles in the environment and target based hybrid singulation need to be considered.

Author Contributions: Conceptualization: B.-J.Y., M.U.A.K. and J.Y.L.; methodology: M.U.A.K., A.I. and B.-J.Y.; software: M.U.A.K., H.H., A.I. and S.K.; writing—original draft preparation: M.U.A.K.; writing—review and editing: M.U.A.K., J.Y.L. and B.-J.Y. supervision: B.-J.Y., J.Y.L. and S.L.; experimentation: S.K. and M.U.A.K. All authors have read and agreed to the published version of the manuscript.

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

Abbreviations

a	Actuated joint
e	Coefficient of Restitution
e_{o-o}	Coefficient of Friction Between Objects
e_{o-m}	Coefficient of Friction Between Object and Manipulator
F_{ext}	Effective Force Vector
g_a	Gravity Load
HSV	Hue, Saturation, Value
I_a	Inertia of Body A
$[I_{aa}]$	Inertia Matrix
m_a	Mass of Body A

n	Normal Vector
PCA	Principal Component Analysis
$[P_{aaa}]$	Inertial Power array
RGB	Red Green Blue
T	Transpose of Matrices and Vectors
v_I	Velocity of Robot
v_{env}	Velocity of Robot
v_a	Mass Center Velocity of Body A
$\overline{v'}_a$	Post Impact Velocity of Body A
ω_a	Angular velocity of Body A
μ_{o-T}	Coefficient of Friction Between Object and Table
τ_a	Joint Torque

References

1. Dogar, M.; Srinivasa, S. A framework for push-grasping in clutter. In Proceedings of the Robotics: Science and Systems VII, University of Southern California, Los Angeles, CA, USA, 27–30 June 2011; Volume 1.
2. Kelley, R.B.; Martins, H.A.; Birk, J.R.; Dessimoz, J.-D. Three vision algorithms for acquiring workpieces from bins. *Proc. IEEE* **1983**, *71*, 803–820.
3. Jacobson, L.; Wechsler, H. Invariant image representation: A path toward solving the bin-picking problem. In Proceedings of the IEEE International Conference on Robotics and Automation, Atlanta, GA, USA, 13–15 March 1984; Volume 1, pp. 190–199.
4. Cosgun, A.; Hermans, T.; Emeli, V.; Stilman, M. Push planning for object placement on cluttered table surfaces. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Francisco, CA, USA, 25–30 September 2011; pp. 4627–4632.
5. King, J.E. Robust Rearrangement Planning Using Nonprehensile Interaction. Ph.D. Dissertation, Carnegie Mellon University, Pittsburgh, PA, USA, 2016.
6. Scholz, J.; Levihn, M.; Isbell, C.; Wingate, D. A physics-based model prior for object-oriented mdps. In Proceedings of the International Conference on Machine Learning, Beijing, China, 21–26 June 2014; pp. 1089–1097.
7. Marios, K.; Sotiris, M. Robust object grasping in clutter via singulation. In Proceedings of the IEEE International Conference on Robotics and Automation, Montreal, QC, Canada, 20–24 May 2019; pp. 1596–1600.
8. Hermans, T.; Li, F.; Rehg, J.M.; Bobick, A.F. Learning contact locations for pushing and orienting unknown objects. In Proceedings of the 13th IEEE-RAS International Conference on Humanoid Robots, Atlanta, GA, USA, 15–17 October 2013; pp. 435–442.
9. Kopicki, M.; Zurek, S.; Stolkin, R.; Mörwald, T.; Wyatt, J. Learning to predict how rigid objects behave under simple manipulation. In Proceeding of the IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 5722–5729.
10. Zhu, S.; Kimmel, A.; Bekris, K.E.; Boularias, A. Fast model identification via physics engines for data-efficient policy search. *arXiv* **2017**, arXiv:1710.08893.
11. Pinto, L.; Gupta, A. Supersizing self-supervision: Learning to grasp from 50 k tries and 700 robot hours. In Proceeding of the IEEE International Conference on Robotics and Automation, Stockholm, Sweden, 16–21 May 2016; pp. 3406–3413.
12. Levine, S.; Pastor, P.; Krizhevsky, A.; Ibarz, J.; Quillen, D. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. J. Robot. Res.* **2018**, *37*, 421–436. [CrossRef]
13. Katz, D.; Pyuro, Y.; Brock, O. Learning to manipulate articulated objects in unstructured environments using a grounded relational representation. In *Robotics: Science and Systems*; MIT Press: Cambridge, MA, USA, 2008.
14. Katz, D.; Brock, O. Extracting planar kinematic models using interactive perception. In *Unifying Perspectives in Computational and Robot Vision*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 11–23.
15. Imran, A.; Kim, S.-H.; Park, Y.-B.; Suh, I.H.; Yi, B.-J. Singulation of Objects in Cluttered Environment Using Dynamic Estimation of Physical Properties. *Appl. Sci.* **2019**, *9*, 3536. [CrossRef]

16. Walker, I.D. Impact configurations and measures for kinematically redundant and multiple armed robot systems. *IEEE Trans. Robot. Autom.* **1994**, *10*, 670–683. [CrossRef]

17. 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]

18. Available online: http://www.virtualmotion.co.kr/ (accessed on 8 December 2019).

19. Dogar, M.R.; Srinivasa, S.S. A planning framework for non-prehensile manipulation under clutter and uncertainty. *Auton. Robot.* **2012**, *33*, 217–236. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.