Robust Perception of Stiffness of Deformable Objects Without Using, Vision, Force or Tactile Sensors

I Ishrath[1], Harish J. Palanthandalam-Madapusi[1]

***Abstract*— This paper presents a robust approach for in-grasp stiffness estimation of deformable objects for robotic hands without the use of vision, force, or tactile sensors. Traditional methods of estimating the stiffness of objects often rely on expensive sensors. In contrast, the proposed method aims to estimate the stiffness of an object being grasped by a robotic hand by executing a quick gentle deformation of the object (similar to how we humans test whether a fruit is ripe or not) and using joint-level position and torque sensing only. The proposed method utilizes a momentum-based disturbance observer for force estimation and the Recursive Least Squares (RLS) algorithm with an exponential forgetting for real-time robust stiffness estimation. While conceptually relatively straightforward, we demonstrate robust performance of the proposed method through simulations and experiments with 6 deformable objects (Orange, Potato, Tomato, Mango, Lemon, and Sponge Ball) and with a low-cost LEAP hand. The results are validated against true stiffness values estimated using a separate experimental test rig. The proposed method yield estimates that are accurate to within 0.005 N/mm of the true stiffness values in all cases, with the percentage errors being higher for softer objects. The target deformation in all tests is 5-10mm with the stiffness estimates converging in most cases within the first 2-3 mm of deformation of the object. The results demonstrate that the developed framework can effectively estimate the stiffness of various objects, promising reliable and accurate manipulation capabilities. This work offers a practical and economical solution for enhancing robotic manipulation in applications involving deformable object with unknown properties and is immediately generalizable to a wide range of graspers and grasping configurations.**

Robotic manipulation, Deformable objects, LEAP hand, Stiffness estimation, Recursive Least Squares, Momentum-based disturbance observer.

I. INTRODUCTION

With a surge in development of humanoid robots particularly targeting unstructured environments such as domestic applications, there is an immediate need for robots to perceive the stiffness of objects (such as fruits and food items) for their safe handling. In addition to domestic applications, other applications including fruit picking, food handling industry, sorting in manufacturing, space robots in unknown environments, medical applications, and food serving in hospitality demand versatile and reliable grasping of deformable objects. In these applications, the inherent variability and unpredictability of deformable objects can pose challenges to traditional robotic manipulation techniques unless the manipulation strategy is fine-tuned a-priori for that particular object. Estimating the stiffness of in-grasp objects is crucial in many such applications.

Fig. 1: Illustration of LEAP hand grasping an Orange

Traditional approaches to estimate stiffness often rely on expensive sensors, as discussed in [1], where an object parameter-based method is proposed for gripping tasks. Recent advancements have introduced more accessible approaches utilizing different sensors and estimation techniques. For instance, [2] employs tactile and piezoresistive sensors along with compact feed-forward neural networks to determine the nonlinear relationship between sensor readings and the actual stiffness of fruits. Another study [3] proposes a framework that estimates a probabilistic distribution of an object's stiffness using visual observation and contact information based on object detection and Gaussian mixture models (GMM), and is validated through object identification tasks. Additionally, a trainable system using Inertial Measurement Units (IMUs) on a Yale OpenHand soft gripper is presented in [4], reliably estimating object stiffness from interaction signals. The Candidate Observer Based Algorithm (COBA) in [5] employs dual force observers to estimate target object stiffness online. In medical applications, [6] demonstrates a fast, online technique for estimating organ shape and stiffness, achieving near video-frame-rate updates. Finite Element Methods (FEM) are also leveraged in [7] to estimate the elasticity parameters of deformable objects

[1]I. Ishrath and H. J. Palanthandalam-Madapusi are with the IITGN Robotics Laboratory, Dept. of Mechanical Engineering, Indian Institute of Technology Gandhinagar, GJ - 382424, India.

Email: ishrathishrath@iitgn.ac.in

Funding: The authors gratefully acknowledge support from Gujarat Council of Science and Technology through grant number GUJCOST/STIR&D/23-24/1711 for this work.

Corresponding author is I Ishrath.

through combined visual and force data, eliminating the need for force sensors. These techniques find applications in robotic grasping of objects of unknown weight, friction, and stiffness, as demonstrated in [8], which devises a methodology for setting grasping force without prior knowledge of an object's characteristics.

In this work we take inspiration from the fact that humans in a supermarket or at home are able to pick up a fruit (or any other deformable object) and perceive the stiffness quickly by executing a small gentle deformation of the fruit and using the proprioception feedback (even with no visual input). We therefore strive to develop a simple, robust, and cost-effective method for estimating object stiffness by executing a quick gentle object deformation of the object while in grasp of a humanoid hand and without using any visual or force or tactile sensing. We combine a momentum-conservation based disturbance observer to estimate real-time contact forces during grasping, and a Recursive Least Squares (RLS) algorithm for stiffness coefficient estimation (based on modelling the object as an elastic element) of the object. The developed method is applicable to a wide range of graspers and grasping configurations. We further demonstrate the method both in simulation and implementation on a low-cost hand, the LEAP hand. We validate the efficacy of the method by validating it against estimated true stiffness for six different deformable objects in experiments. The proposed approach offers a practical and economical solution for real-time estimation of in-grasp deformable objects, facilitating enhanced robotic manipulation capabilities in various real-world applications.

II. Methodology

The approach taken is for the robotic hand to hold the object in a two-finger pinch grasp and execute a gentle object deformation (by a few mm) and use disturbance observer to estimate the contact forces, use joint angle sensors to estimate the displacements, and use RLS to estimate the object stiffness.

We start by discussing the object modelling, outlining the theoretical framework and mathematical models used. Next, we describe the robotic hand setup, including the hardware components and control methods. We then detail the disturbance observer, explaining its design and implementation for handling external disturbances. Finally, we summarise the Recursive Least Squares (RLS) algorithm, focusing on its use for force estimation and stiffness estimation.

A. Object modelling

Numerous models have been proposed in the literature to approximate the dynamics of deformable objects as mentioned in [9]. The Burger model, in particular, has been extensively cited and applied due to its ability to capture complex viscoelastic behaviors [10]. In our study, however, we adopt a simpler approach by modelling the object as a stiffness element. Furthermore, inertia effects can be neglected by controlling the gentle object deformation execution such that the object's center of mass undergoes negligible acceleration. With this assumption, the relationship between the external force and the object deformation can be expressed as

$$F_{ext} - Kx = 0, \tag{1}$$

where F_{ext} is the external force, K is the stiffness and X is the deformation of the object.

B. Modelling LEAP Hand fingers

The LEAP Hand [11] is an affordable, dexterous, and anthropomorphic robotic hand. The LEAP hand has 3 fingers and a thumb, with each finger of the hand having 4 DOF as shown in fig 1. One of the joints enables the abduction and adduction mechanism. However, to restrict the motion of the finger to a plane, we fix this joint in position control mode for the remainder of this work. Moving ahead with this constraint, each finger can be modelled as a planar 3R serial chain manipulator.

1) Kinematics: The kinematic equations of the finger are given as below

$$P_x = L_1 \cos(q_1) + L_2 \cos(q_1 + q_2) + L_3 \cos(q_1 + q_2 + q_3), \tag{2}$$

$$P_y = L_1 \sin(q_1) + L_2 \sin(q_1 + q_2) + L_3 \sin(q_1 + q_2 + q_3), \tag{3}$$

where q_1, q_2 and q_3 are joint angles (relative) and L_1, L_2 and L_3 are lengths of finger segments (Figure 2). This forms a redundant system and multiple finger configurations are possible to reach a desired contact point. Hence, we apply an additional constraint on the end-tip segment of the finger to resolve the redundant system as

$$q_1 + q_2 + q_3 = 0. \tag{4}$$

The constraint in (4), along with (2), (3) gives a unique solution for q_1, q_2 an q_3 along the points P_x, P_y for a given trajectory.

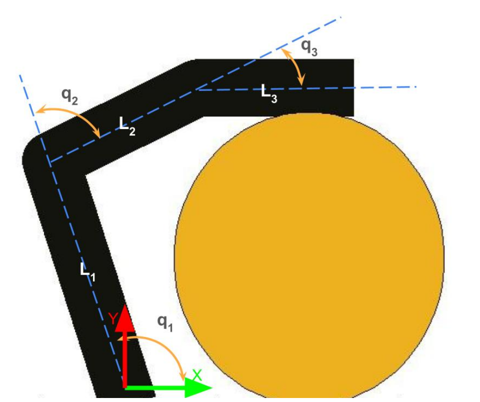


Fig. 2: Schematic diagram of finger

2) Dynamics: The dynamic equations of the finger are derived and expressed in the manipulator form as [12]

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = T_c + T_{ext}, \tag{5}$$

where M, C and G are the mass, Coriolis and gravity matrices for the 3R system. The expression for these are derived using the Euler-Lagrange formulation [12] and are omitted here in the interest of space.

3) Control Law: To execute the two-finger gentle object deformation, one of the fingers (the thumb in this case) is commanded to hold position while the other finger tracks a desired trajectory in task space (typically a displacement of a few mm along vertical axis). For this trajectory tracking, the control law is formulated using the feedback linearization method [12].

Using the dynamics of finger, the torque input at each joint using joint accelerations $\ddot{q}$, error in displacement, e and error in velocity, $\dot{e}$ is given as

$$e = q_d - q, \tag{6}$$

$$\dot{e} = \dot{q}_d - \dot{q}, \tag{7}$$

$$T_c = M(q) \times (\ddot{q}_d + K_p e + K_i \int e(t)dt + K_d \dot{e}) \tag{8}$$

$$+ C(q,\dot{q})\dot{q} + G(q) - T_e, \tag{9}$$

where K_p and K_d are proportional and derivative gains and T_e is torque at each joint due to external force estimated using Disturbance observer as mentioned in section II-C. The gains K_p, K_i, and K_d are chosen to place the roots of the characteristic equation for desirable transient response performance. In practice K_i is often chosen to be zero [12].

C. Force Estimation using Momentum based Disturbance Observer

The fundamental principle behind this approach is the use of the principle of conservation of momentum, which allows the observer to infer disturbances based on deviations from expected system behavior. In this section, we design an observer to estimate the joint torques and use these to compute the end-tip forces.

The observer starts with a dynamic model of the robotic system, which includes its mass, inertia, and the equations of motion. The framework developed in [13] and adopted in [14] is used. As the system operates, sensors (typically encoders) measure the motors actual angular positions. These measurements are used to calculate the real-time momentum of the system as

$$P_i = M(q)\dot{q}, \tag{10}$$

where M is the inertia matrix of the finger. Next, we predicts the expected momentum of the robot, under given control inputs (such as torques or forces applied by the actuators) using:

$$e_P = P_i - \hat{P}_{i-1}, \tag{11}$$

$$\dot{\hat{P}}_i = (T_c + C(q,\dot{q})\dot{q} - G(q) + K_m e_P), \tag{12}$$

where T_c is the control input, C is the Coriolis matrix, e_P is the error between real-time momentum and predicted momentum and K_m is the gain matrix. The observer compares the predicted momentum from the dynamic model with the actual measured momentum. Any discrepancy between these two values is indicative of an external disturbance force acting on the system. The torques at each joint can be estimated as

$$T_e = K_m(P_i - \hat{P}_i), \tag{13}$$

$$F_e = J^{-1} T_e, \tag{14}$$

where K_m is a diagonal gain matrix with positive gains, P_i and $\hat{P}_i$ are real-time and estimated momentum respectively and J is the Jacobian of the finger. Higher the gain values, more accurate is the disturbance estimation and higher is the sensitivity to noise. Since in practice the signals are noisy, lower values for gains in K_m is preferred [13].

D. Parameter Estimation using Recursive Least Squares with exponential forgetting (RLS) method

The Recursive Least Squares (RLS) method is a robust and efficient algorithm for estimating parameters in a dynamic system in real-time. We use this algorithm to estimate the stiffness of the object using the external forces estimated in section II-C

The RLS algorithm is initialized with an initial estimate of the stiffness parameter $\hat{K}_k$ and an initial value for the error covariance matrix P_0, which quantifies the confidence in the initial estimate. Using these initial conditions and framework outlined in [15], the external force is estimated as mentioned below

$$\hat{F}_e = \hat{K}_k \psi^T(t). \tag{15}$$

A weighing matrix Q is computed using the Covariance matrix P and the deformation ψ as shown below

$$Q(t) = P_{k-1}\psi(t)(\lambda + \psi^T(t)P_{k-1}\psi(t))^{-1}, \tag{16}$$

$$P_k = \frac{(I - Q(t)\psi^T(t))P_{k-1}}{\lambda}, \tag{17}$$

where λ is the forgetting factor and P is a positive definite matrix.

Using the error in force estimation and using (14) and (15) and the gain matrix Q, the estimated stiffness parameter is updated as

$$e = F_e - \hat{F}_e, \tag{18}$$

$$\hat{K}_k = \hat{K}_{k-1} + Qe. \tag{19}$$

At each time step k, the RLS algorithm updates the estimate of the stiffness parameter. Over time, as more data is processed, the stiffness estimate $\hat{K}_k$ converges to the true stiffness parameter assuming the model and data are accurate.

III. Simulations and Experiments

In this section we describe the experiment carried out to estimate the true stiffness of 6 different objects - Lemon, Mango, Orange, Potato, Tomato and Sponge Ball (Figure 3). We also describe the simulations and experiments performed using the proposed method on the LEAP hand to estimate the stiffness of these 6 objects in real-time.

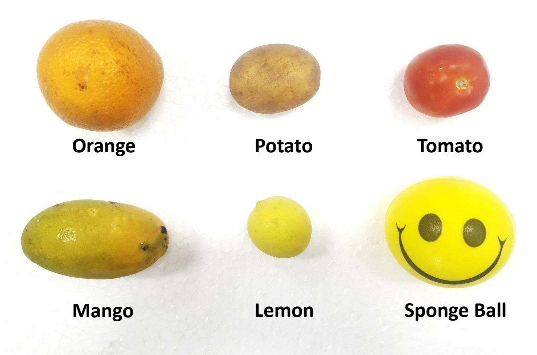


Fig. 3: The six deformable objects considered in this work for stiffness estimation.

A. Experimental True Stiffness Computation

The experimental stiffness test rig setup consists of a base and top platform between which the deformable object is placed as shown in Figure 4. The top platform is actuated through a lead screw (Pitch 2mm) which is driven by a Dynamixel XL430-W250-T motor at certain velocity (15rpm). A Futek single axis force sensor is mounted on the top plate. After some iterations, a conical contact tip geometry was devised and employed in all experiments to ensure point contact and to avoid contact of the object with the frame (which would lead to erroneous force sensor readings as some component of the contact forces would be directly transmitted to the frame). A maximum downward displacement of 10 mm is given to the top platform and the corresponding forces are recorded. Figure 5 shows the forces recorded and the obtained true stiffness for each object.

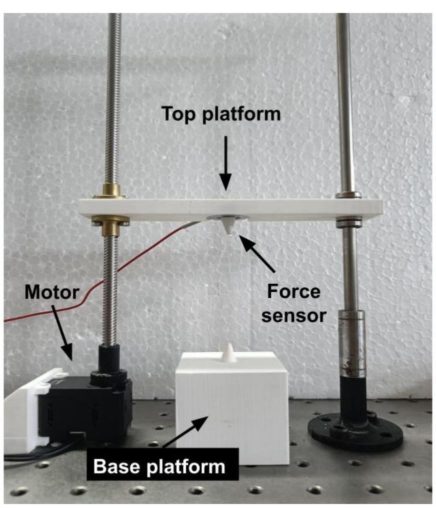


Fig. 4: Experimental test rig designed to estimate the true stiffness of deformable objects.

TABLE I: True stiffness of various deformable objects as estimated usign the experimental test rig.

S. No.	Object	True Stiffness (N/mm)	Diameter (mm)
1	Lemon	0.079	36
2	Mango	0.034	46
3	Orange	0.075	61
4	Potato	0.119	37
5	Tomato	0.036	40
6	Sponge Ball	0.023	70

The stiffness values obtained in Table I are further used in the subsequent sections to simulate and experimentally validate the stiffness estimation algorithm.

B. Simulations and Experiments using LEAP Hand

This section describes the simulations and experiments carried out using the LEAP hand. The LEAP hand [11] is an affordable platform for implementing control strategies and testing algorithms. Each finger is actuated by Dynamixel XL330-M288-T motors, which support five operational modes: Position, Current-based Position, Velocity, Current, and Extended Position. In particular current sensors integrated with the motors provide current readings from which joint torques are estimated. Advanced control strategies can then be built upon these modes. The whole setup cost us less than \$ 1000 to build, thus serving as an ideal platform for demonstrating our method of estimating stiffness without using any expensive sensors. Both for simulations and experiment, the 6 objects mentioned in Table I are deformed by a specified value along the Y-axis. Using the equations derived in section II, a computed torque is given to drive each motor of the finger based on the force estimation using disturbance observer. The stiffness of the objects are then estimated on-line using the RLS algorithm described in section II-D.

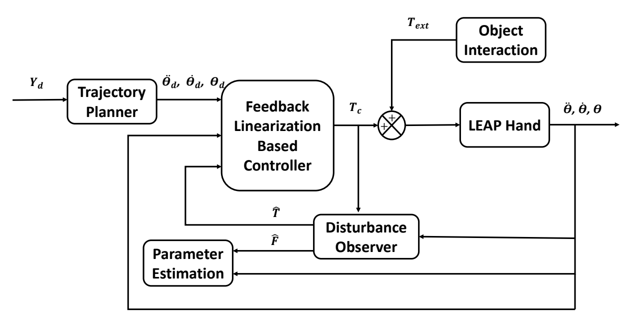
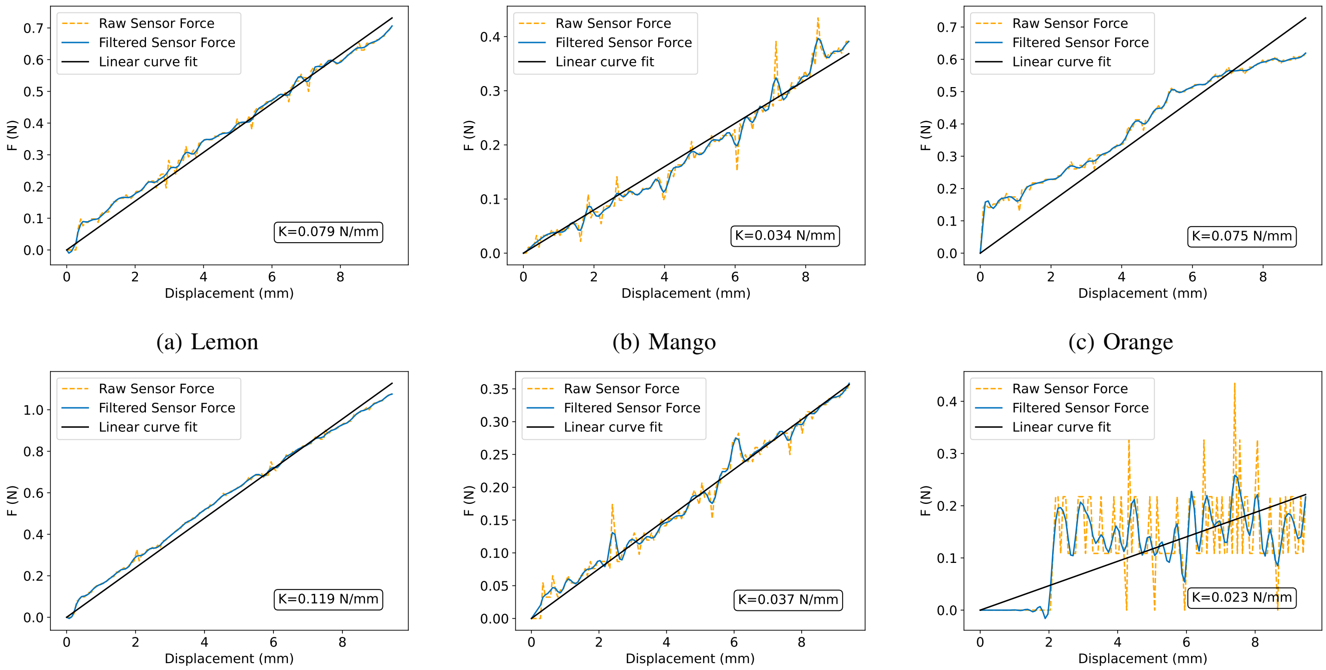


Fig. 6: Block diagram illustrating the stiffness estimation and control architecture.

1) Simulations: Using the modelling approach as mentioned in section II and true stiffness values from Table I, a MATLAB code is used to track a cubic trajectory



(a) Lemon (b) Mango (c) Orange

(d) Potato (e) Tomato (f) Sponge

Fig. 5: True Stiffness Experiment: True stiffness of different objects as estimated using the experimental test rig.

such that the objects are deformed by a maximum of 5mm along the Y-axis. A time step of 0.02s and a forgetting factor of 0.5 is used. Starting from a stable two-point grasp configuration, the corresponding actual displacement and stiffness estimation are recorded at every time step.

The Figure 7 shows the trajectory tracked by the finger tip and the stiffness estimation with respect to time. From the results, we see that deformation of 5mm is achieved in all objects and the estimated stiffness converges to a constant value after 4s when the object is deformed by at least 3.75mm.

The estimated stiffness for each object in simulations are mentioned in Table II

2) Experiments - LEAP Hand: Using the modelling approach as mentioned in section II and true stiffness values from Table I, a python script is used to command the joint angles to each motor of the LEAP hand finger to track a cubic trajectory such that the objects are deformed by a maximum of 10mm along the Y-axis. The object is placed between the thumb and the finger tip forming a stable two-point grasp, as assumed in the simulation section. The motor positions of the thumb are frozen and hence it acts like a rigid support throughout the experiment. While tracking the trajectory, corresponding joint torques and positions of the finger are recorded, which are further used to estimate external force (14) and stiffness of the object. Additionally, a single axis force sensor is mounted on the finger as shown in Figure 8 to validate the estimated external force (but not used in the proposed method for estimating stiffness). The joints angles, torques and the force sensor data are all collected at a rate of 50Hz and a forgetting factor of 0.5 is used to estimate stiffness.

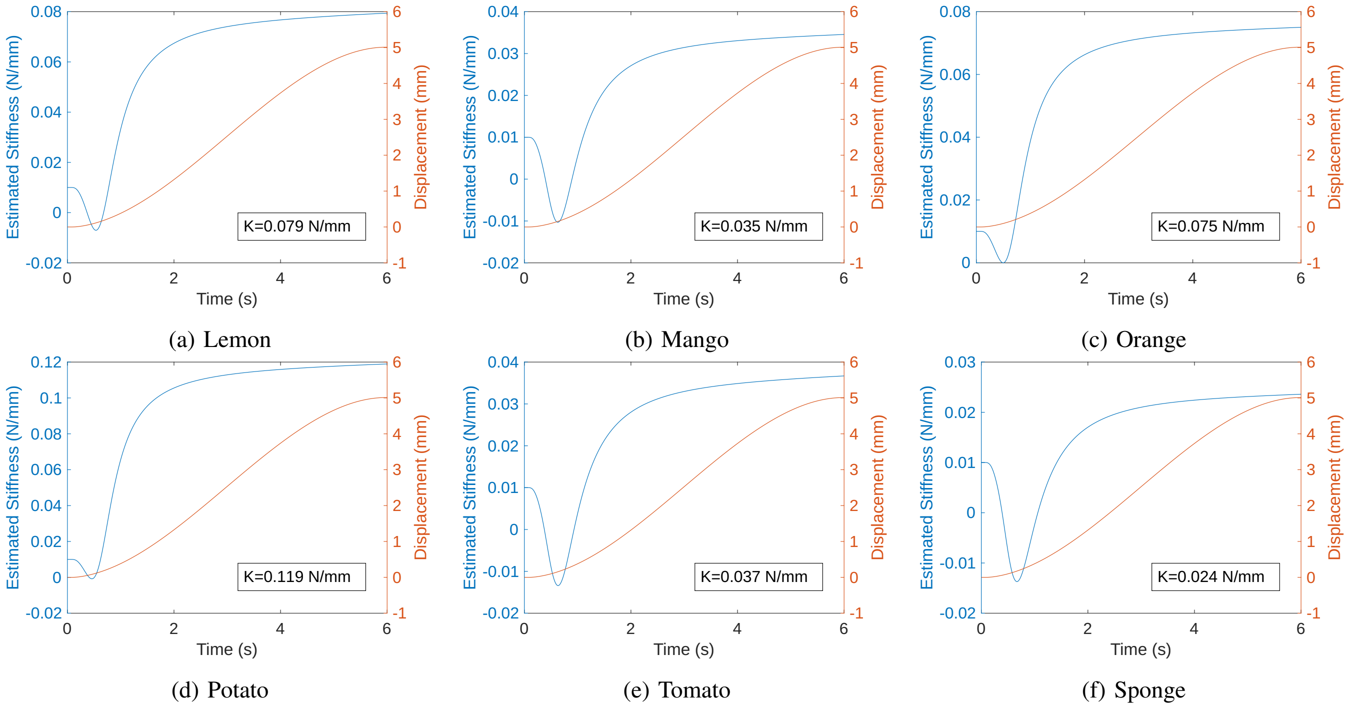
Figure 9 shows two plots for each object, the variation of sensor and estimated force with deformation and estimation of stiffness using RLS algorithm while the object is being deformed with respect to time. From the force plots, a deviation of maximum 0.2N is observed between the forces recorded by the sensor and those estimated using (14). The stiffness values of different objects obtained by fitting a linear curve along the estimated force are mentioned in Table II.

From the plots showing the stiffness estimation using RLS, we see that the actual deformation is different for different objects as one would expect. Owing to the limitations on motors torque capacity, the commanded trajectory cannot be tracked as seen in all objects, except Sponge Ball. Also, we see that the estimated stiffness values converge to a constant value at 4s where the object is at least deformed by 50% of the maximum deformation achieved.

The estimated stiffness for each object by linearly fitting curve on estimated force and by using the RLS algorithm are summarised in Table II

IV. Results and Discussion

In this section we make some key observations and draw insights from the results obtained in previous sections. Based on the true stiffness values obtained in Table I, the objects can be classified into relatively softer and stiffer objects with Lemon, Orange and Potato being stiffer and the other three being softer. From the results obtained from simulations and experiments for the real-time estimation of stiffness for different objects considered, the following insights are drawn



(a) Lemon (b) Mango (c) Orange

(d) Potato (e) Tomato (f) Sponge

Fig. 7: Simulation Results: Estimated stiffness of different object during the deformation

TABLE II: Comparision of estimated stiffness of various deformable objects

S. No.	Object	True Stiffness	Simulations		Using Force Sensor		Proposed Algorithm	
		K (N/mm)	K (N/mm)	Error (%)	K (N/mm)	Error (%)	K (N/mm)	Error (%)
1	Lemon	0.079	0.079	0	0.078	-1.26	0.077	-2.53
2	Mango	0.034	0.035	2.94	0.035	2.94	0.036	5.88
3	Orange	0.075	0.075	0	0.079	5.33	0.079	5.33
4	Potato	0.119	0.119	0	0.121	1.68	0.120	0.84
5	Tomato	0.036	0.037	2.77	0.041	13.88	0.041	13.88
6	Sponge Ball	0.023	0.024	4.34	0.026	13.04	0.026	13.04

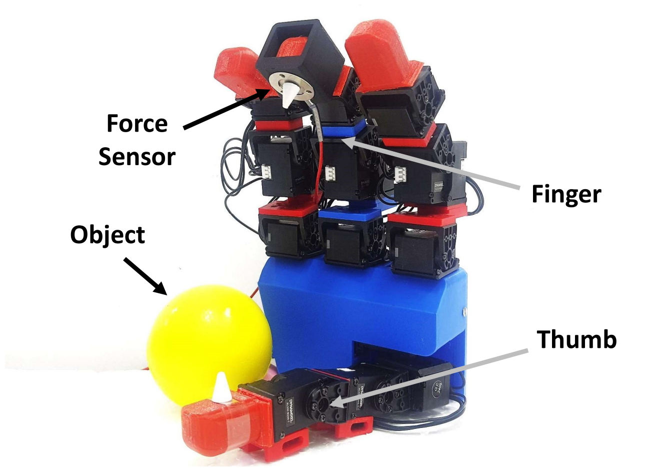
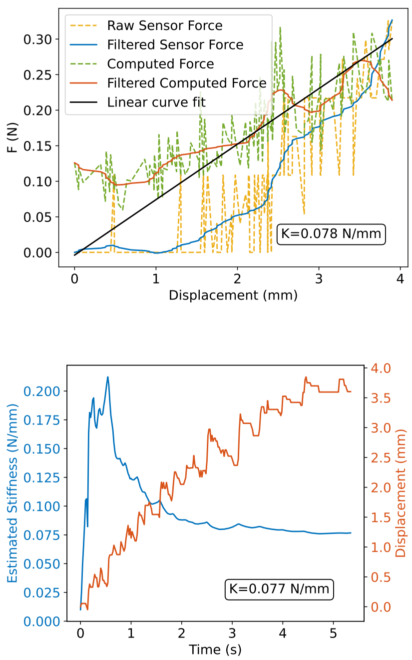
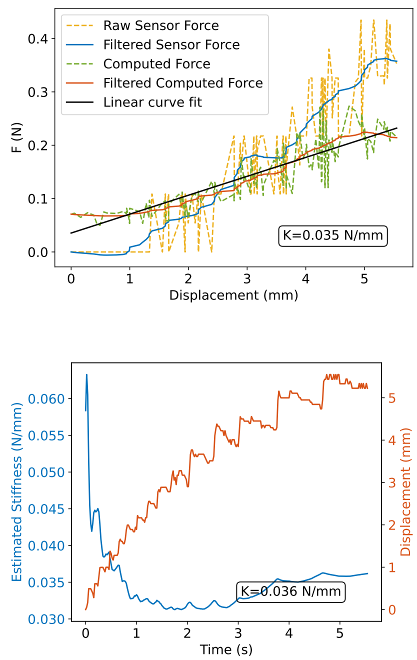


Fig. 8: LEAP hand setup with force sensor mounted (for validation only). The stiffness of the object being held is estimated using joint torque data from current sensing at the motors and using the proposed method with disturbance observer and recursive least-squares estimation algorithms.

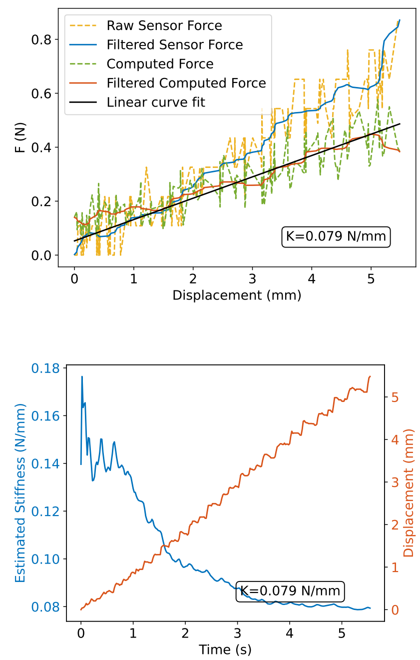
1) Accuracy of force estimation - Accurate estimation of forces at the end-tip are essential to compute the stiffness of the objects. From Figure 9, we observe that the filtered computed force and the filtered sensor force are closely aligned with a maximum deviation of 0.3N, as seen in the case of sponge ball.
2) Accuracy of stiffness estimation - From Table II, it is observed that the stiffness estimation in simulations match closely with the true stiffness values mentioned in Table I for the Lemon, Orange and Potato while there is an error of 0.001 N/mm in estimation of stiffness of other objects. In any of the cases, the error in estimation in experiments is not more than 0.004 N/mm. The stiffness values of the objects computed using linear curve fitting are observed to be greater than the estimates obtained in simulation, with Lemon being an exception. It is to be noted that the linearly fit curve doesn't always pass through the origin, especially in the case of softer objects, implying a small initial deformation that leads to some compressive force readings from sensor. The proposed algorithm on the other hand, produced estimates lower or equal to the linearly fit curve stiffness values for stiffer objects.



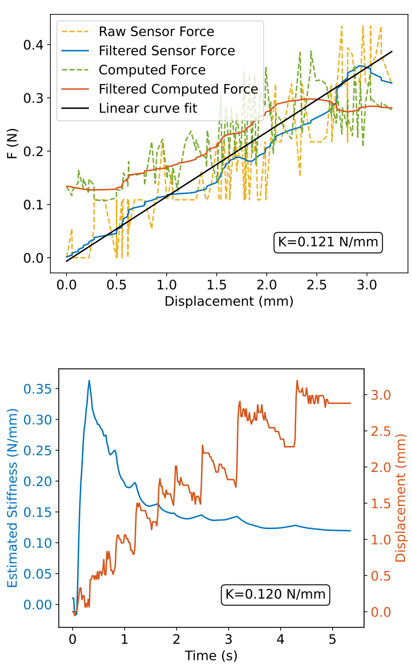
(a) Lemon



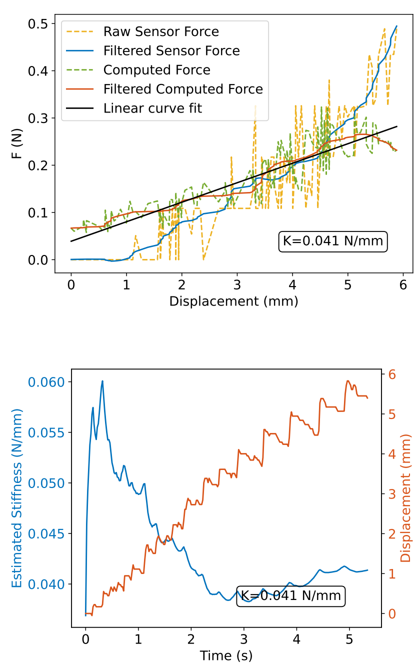
(b) Mango



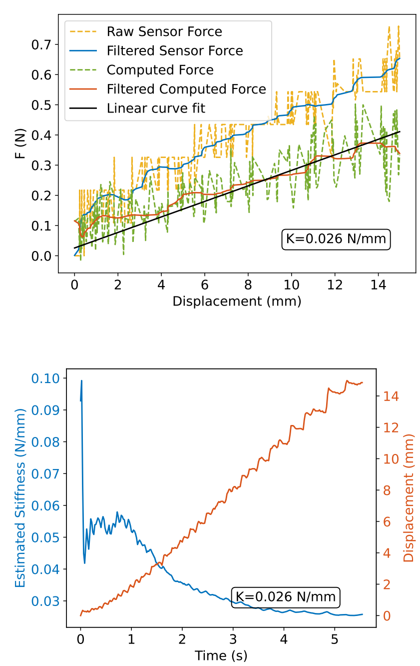
(c) Orange



(d) Potato



(e) Tomato



(f) Sponge

Fig. 9: Experimental Results: Estimated and actual stiffness of different object during deformation using the LEAP hand.

The estimates for the softer objects, Mango, Tomato and Sponge are observed to be greater than or equal to true stiffness estimates in all three cases (Simulation, Force sensor and Proposed algorithm). A maximum of 13.88% of error is observed in the estimates, with softer objects having the higher errors.

3) Perception time - While the Linear curve fitting would require collecting and storing data, followed by offline processing, the RLS algorithm is observed to quickly converge to the true stiffness estimates. From the plots in Figure 7, we observe that starting at 2s, the algorithm starts to converge to a constant stiffness value. This minimum time required to perceive the stiffness of the object is observed to be slightly higher, 2.5s during experiments as seen the the fig. 9. This response time depends on the planned trajectory, forgetting factor, controller and actuator frequencies.
4) Grasp configuration - We assumed a simple two point contact model to draw insights. However, the developed algorithm is easily generalised and can be applied to multi-contact grasp by planning the trajectory for one of the fingers, hence restricting the deformation to a single axis. Contact modelling plays an essential role in achieving this. With a surface contact, the object no longer undergoes axial deformation which poses a challenge of estimating the stiffness. Hence, ensuring a point contact during deformation is essential.

V. Conclusions

The developed framework provides a robust, simple and effective solution to estimating stiffness of a grasped object without the need for vision, force or tactile sensors. The method using a disturbance observer and RLS approach is straightforward to implement in various configurations and situations. The results obtained and discussed in the previous section highlight the robust performance of the developed framework to perceive stiffness of deformable objects in real-time. With only an error of not more than 0.005 N/mm in estimation of stiffness, the developed algorithm can grasp objects susceptible to deformation with least damage. The quick response of 2s in estimating the stiffness, further ensures minimal applied forces to be secure a grasp and manipulate. In future, the algorithm can be extended to estimate other object parameters based on the model chosen for the object. Also, an elaborate analysis on grasp stability and slip during deformation would be significant contributions to enable manipulation of the object.

VI. Acknowledgement

We thank Barat S, a PhD scholar and Mihika Desai, an undergraduate student in IIT Gandhinagar for their contribution to help setup the simulations and experiments.

VII. Appendix

References

[1] C. Wang, X. Zang, X. Zhang, Y. Liu, and J. Zhao, "Parameter estimation and object gripping based on fingertip force/torque sensors," Measurement, vol. 179, p. 109479, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263224121004632

[2] F. E. Erukainure, V. Parque, M. A. Hassan, and A. M. R. FathElbab, "Towards estimating the stiffness of soft fruits using a piezoresistive tactile sensor and neural network schemes," in 2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), 2022, pp. 290–295.

[3] M. Kamigaki and S. Katsura, "Stiffness estimation from vision and touch using object detection and probabilistic model: An application to object identification," in IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society, 2021, pp. 1–6.

[4] M. Bednarek, P. Kicki, J. Bednarek, and K. Walas, "Gaining a sense of touch object stiffness estimation using a soft gripper and neural networks," Electronics, vol. 10, no. 1, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/1/96

[5] F. Coutinho and R. Cortesão, "Online stiffness estimation for robotic tasks with force observers," Control Engineering Practice, vol. 24, pp. 92–105, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967066113002062

[6] P. Chalasani, L. Wang, R. Yasin, N. Simaan, and R. H. Taylor, "Preliminary evaluation of an online estimation method for organ geometry and tissue stiffness," IEEE Robotics and Automation Letters, vol. 3, no. 3, pp. 1816–1823, 2018.

[7] A. Sengupta, R. Lagneau, A. Krupa, E. Marchand, and M. Marchal, "Simultaneous tracking and elasticity parameter estimation of deformable objects," in 2020 IEEE International Conference on Robotics and Automation (ICRA), 2020, pp. 10 038–10 044.

[8] T. Sugaiwa, G. Fujii, H. Iwata, and S. Sugano, "A methodology for setting grasping force for picking up an object with unknown weight, friction, and stiffness," in 2010 10th IEEE-RAS International Conference on Humanoid Robots, 2010, pp. 288–293.

[9] V. E. Arriola-Rios, P. Guler, F. Ficuciello, D. Kragic, B. Siciliano, and J. L. Wyatt, "Modeling of deformable objects for robotic manipulation: A tutorial and review," Frontiers in Robotics and AI, vol. 7, p. 82, 2020.

[10] W. Ji, Z. Qian, B. Xu, G. Chen, and D. Zhao, "Apple viscoelastic complex model for bruise damage analysis in constant velocity grasping by gripper," Computers and Electronics in Agriculture, vol. 162, pp. 907–920, 2019.

[11] K. Shaw, A. Agarwal, and D. Pathak, "Leap hand: Low-cost, efficient, and anthropomorphic hand for robot learning," Robotics: Science and Systems (RSS), 2023.

[12] K. M. Lynch and F. C. Park, Modern Robotics: Mechanics, Planning, and Control, 1st ed. USA: Cambridge University Press, 2017.

[13] M. Van Damme, P. Beyl, B. Vanderborght, V. Grosu, R. Van Ham, I. Vanderniepen, A. Matthys, and D. Lefeber, "Estimating robot end-effector force from noisy actuator torque measurements," in 2011 IEEE International Conference on Robotics and Automation, 2011, pp. 1108–1113.

[14] S. Jadav and H. J. Palanthandalam-Madapusi, "Configuration and force-field aware variable impedance control with faster re-learning," Journal of Intelligent & Robotic Systems, vol. 110, no. 1, p. 3, 2024.

[15] K. J. Astrom and B. Wittenmark, Adaptive Control, 2nd ed. USA: Addison-Wesley Longman Publishing Co., Inc., 1994.