ICAR '97
Monterey, CA, July 7-9, 1997

Recursive Learning for Deformable Object Manipulation

Ayanna M. Howard, George A. Bekey
Institute for Robotics and Intelligent Systems, School of Engineering
University of Southern California, Los Angeles, California 90089

Abstract

This paper presents a generalized approach to handling of 3D deformable objects. Our task is to learn robotic grasping characteristics for a non-rigid object represented by a physically-based model. The model is derived from discretizing the object into a network of interconnected particles and springs. Using Newtonian equations, we model the particle motion of a deformable object and thus calculate the deformation characteristics of the object. These deformation characteristics allow us to learn the required minimum forces necessary to successfully grasp the object and by linking these parameters into a learning table, we can subsequently retrieve the forces necessary to grasp an object presented to the system during run time. This new method of learning is presented and the results of a virtual simulation are shown.

Keywords: Robotic Grasping, Learning, Deformation, Elasticity

I. Introduction

Most robotic systems have been built under the assumption that manipulation of rigid objects remains the primary task. Geometry of the object is usually very static, with very little variance between one instance of the object and another. The robot must have knowledge pertaining to the exact structure and location of objects in the environment and the precise actions to be performed. But, in reality, many objects are non-rigid. Most are unsymmetrical, compliant, and have alterable shapes. Even solid objects can deform when the object's dimensions become extensive. Seemingly simple manipulations and actions can result in complex changes to the object due to the large deformation caused by external forces. In general, deformable objects may be one-,two-, or three-dimensional (flexible beam, flexible plate, flexible body). In the real world, these classes of objects include balls, beams, hoses, cloth, and wire. Manipulation of a deformable three dimensional object is the least addressed in research areas, with many feeling that ,by addressing the simpler one-dimensional case, a generalized three dimensional solution can be easily found. Unfortunately, no one has, as of yet, developed a generalized solution to manipulation of 3D deformable objects. As a result, automated handling of soft, flexible 3D objects still needs to be addressed.

This proposal presents a generalized approach to handling of 3D deformable objects. In the past, the operating robotic system required in-depth knowledge of object characteristics in order to successfully complete a manipulation task. These attributes included object dimensions, stiffness/elastic coefficients, and density parameters. Normally when an object is unexpectedly presented to the system, such information as this is not readily available. At most the object's weight may possible by measured as it traverses down an assembly line. A robust robotic system therefore requires more general modeling of an object. We therefore propose to address the issue of a generalized model in terms of a learning process. We propose that the attributes which the system needs to know for grasping a deformable object can be learned off-line for a wide range of three dimensional objects. The attributes learned can then be mapped such that, during run-time, enough relevant attributes can be retrieved to grasp any three dimensional object presented to the system. This approach thus ensures that a robotic system remains robust and able to handle a multitude of three dimensional objects belonging to the class of deformable or rigid bodies.

II. Background

There has been some study on automated handling of non-rigid objects. David W. Meer and Stephen M. Rock [5] at Stanford University researched the development of a controller which can explicitly control the deformation of a flexible object. Although the project was successful in implementing an insertion task in which the object had to be deformed within certain specifications in order to perform the assembly operation, the proposed method possessed some limitations. The system only modeled a single flexible degree of freedom. It required an explicit initial model of the object which included mass/object dynamics and a model of the object's flexibility, and complex sensors were required to measure the position and force of the object.

Patton, Swern, Tricamo, and van der Veen [7] used an adaptive control loop to generate correct tension on a piece of cloth. The adaptive controller was used to successfully control the straightening of a wrinkled piece of cloth. The disadvantages to this research effort were the system only modeled a two dimensional object and the system required explicit knowledge of the initial and desired tension in terms of x, y, and θ.

A.M. Annaswamy (MIT) and D. Seto (Boston University) [1] addressed the issue of manipulation of deformable objects using compliant finger pads. The simulation results showed the system's ability to control the global motion of the object while keeping the deformation of the object at a constant. Unfortunately, the parameters which define an object as deformable must explicitly be provided to the system before successful completion of the task.

Robotic handling of deformable objects is comparable to the handling of rigid objects by soft, deformable robotic fingers. Reznik and Laugier [9] from UC Berkeley, address the issue of real-time dynamic simulation and control of a "virtual" deformable object. A simulation program was successfully able to map the "virtual" deformation of a three dimensional finger as it pressed against a rigid surface. The only disadvantage to this system is that the object's volumetric dimensions, density, and elasticity must be known a prior.

Shimoga (Carnegie Mellon University) and Goldenberg (University of Toronto) [11] modeled a soft finger using the Kelvin model in which a spring and damper are placed in parallel. Using experimentally calculated stiffness and damping coefficients in the Kelvin model, and given the desired impedance parameters, the researchers were able to successfully control the impedance of a soft fingertip. Unfortunately this model is only for the one-dimensional case.

III. Technique

Our main focus is to learn an adequate grasp for a deformable object. To be more specific, our task is to learn what forces a multiple robotic mechanism must exert in order to grasp a common deformable object cooperatively. We choose to represent grasping as the act of pushing up against an object from two opposite ends. Our system, therefore, utilizes two cooperative manipulators, each possessing an end-effector constructed as a flat surface palm and possessing a force sensor able to detect and record any force applied by (or against) the palm's surface area.

In order to adequately manipulate an object, we need to maintain a firm grasp on the deformable body. We must therefore maintain an internal representation of the object which keeps track of the dynamic state of the body. We shall begin by focusing on the physical changes of the deformable body. Once a representation for the positional alterations of the deformable object is retrieved, an adequate grasp can be determined. We propose that a relationship between deformation, stiffness, and force can be learned such that an adequate grasp with minimal force can be achieved. Once this relationship is learned, we can utilize these factors to maintain a firm grasp on any deformable body by comparing the current run-time mass and deformability of the object with the learned relationship. When they are equivalent, we can retrieve the necessary force required for grasping a deformable object.

Constructing a Physically Based Model

Discretizing the Object

For a class of elastic, isotropic materials, molecules are organized as an orderly array of atoms arranged in rows such that each atom is at an equilibrium distance from each other. This structural arrangement is called a space lattice and is represented by a 3-dimensional array of cubic blocks [2].

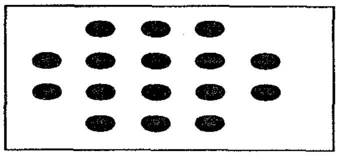

Fig. 1 Space Lattice Arrangement

We shall utilize this characteristic to represent an object as a particle based system constructed from a discretized sampling of its volume. The discretization factor, $\alpha>0$, is derived from the density characteristics inherent to the object's material properties. For this effort, we shall assume all objects presented to the system have known masses. Thus the density can be calculated with respect to the object's mass and volume and can be used for determination of α. Assume the object's perpendicular faces give rise to the (x_0,y_0,z_0) coordinate system and the object's corner is centered at the origin. Given the derived discretization constant, the number of particles that comprise the object can be determined.

$$num_p \approx \frac{V}{\alpha^3} + \sum_{\forall i,j \in x_0,y_0,z_0} \frac{A_{ij}}{\alpha^2} + \sum_{\forall i \in x_0,y_0,z_0} \frac{d_i}{\alpha} + 1$$

where V is the volume of the object, A_{ij} is the area of the object's face projected along axis $\notin$ i,j and fixed at zero, and d_i is the diameter of the object's face projected along axis $\in$ i and fixed at zero (Fig. 2).

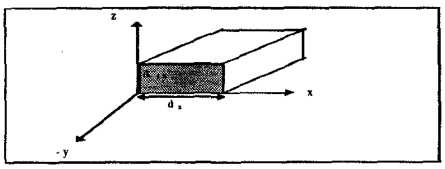

Fig 2. Representation of A_{ij}, d_i

We shall thus represent the particle discretization by ψ_{ijk}, a three dimensional matrix of vectors.

Let R_i = maximum radius of object in i direction

$\forall$ i,j,k $\in$ x,y,z: $f_m = \dfrac{k * (2R_m + \alpha)}{\alpha}$

$$a = [\alpha * (i - f_i)] - R_i$$
$$b = [\alpha * (j - f_j)] - R_j$$
$$c = [\alpha * k] - R_k$$

if $0 \le f(a,b,c) \le g(Ri,Rj,Rk)$, $\psi_{ijk} = (a,b,c)^T$

else, $\psi_{ijk} = \phi$

Modeling Object Deformation

Deformation of a material are caused at the microscopic level by visco-elastic interactions between molecules. Such deformation consists primarily of distortions of the atomic space lattice. When such an object is compressed, the distances between atoms is decreased. Accumulations of these small decreases result in over-all elastic shortenings. The equations governing the particle motion of such a deformable object can be written in Lagrange's form as:

$$m\frac{d^2 p(\psi_{ijk},t)}{dt^2} + \mu\frac{dp(\psi_{ijk},t)}{dt} + \frac{\partial E}{\partial p(\psi_{ijk},t)} = f_p(t)$$

The relationship between distance and applied force is approximately linear for the small changes in space usually considered. Elastic deformation therefore involves volume change and is linear in nature. A viscoelastic material exhibits three basic characteristics: instantaneous elastic response, viscous damping, and delayed elastic response. These characteristics can be modeled by a spring and damper in parallel (Kelvin model). The spring deforms in direct proportion to the amount of load applied. Depending on the rate of loading, the dashpot deforms proportionately, acting as a damper to soak up deformation energy. These two characteristics, the ability to deform in a perfectly elastic manner and the ability to undergo relaxation, enable the Kelvin model to behave visco-elastically. In order to use this model in characterizing a deformable object, we must ensure that there is enough connectivity between mass nodes within the object to adequately model its deformation. In addition, the system must remain as simple as possible by keeping the number of modeling elements to a minimum. The spring and dashpot model is uncomplicated, is able to run in real time, and works well with large deformations. We shall thus utilize the Kelvin model and characterize a deformable object as a set of atomic particles locally interconnected by damped springs (Fig 3). For this case, $p(\psi_{ijk},t)$ is the position of particle ψ_{ijk} at time t, m is the mass of each particle, μ is the damping coefficient, and $\partial E(x)/\partial p$ is the elastic force of the body (the potential energy of the elastic deformation) [12].

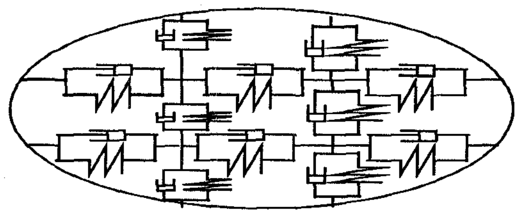

Fig. 3 Network of Particles and Springs

Now, we must determine the deformability of an object. In order to do this, we must determine an approximate value for the spring and damper constant of each spring. The spring constant will be denoted as D - the deformability constant, and μ will denote the damper coefficient. To measure the deformability constant, we use the following procedure:

1. Using both manipulators, apply a known constant force against the object's surface
2. Measure displacement of the surface
3. Calculate D from knowledge of the discretization of the object
4. Repeat this for several force values, plot, and fit to a curve
5. The force-deformation analytical relation is $f_{spring} = D(x)*x$

To measure the damper coefficient, we use the following procedure:

1. Using both manipulators, deform the object surface a known constant displacement
2. Measure force on the manipulator ($f_{damped} = f_{sensed} - f_{spring}$)
3. Repeat this for several displacement values, plot, and fit to curve
4. The force-damping analytical relation is $f_{damped} = \mu(x^`,x)*x^`$

Learning An Adequate Grasp

To learn the characteristics of an adequate grasp, we must determine the relationship between mass, deformability, and force. Once this relationship is learned, we can utilize these factors to maintain a firm grasp on any deformable body by comparing the current run-time mass/deformability of the object with the learned relationship. When they are equivalent, we can retrieve the necessary force required for grasping a deformable object. The steps required to handle manipulation of 3D deformable objects are as follows:

I. Learn what forces a robotic system must exert in order to grasp a deformable 3D object

a) Record dimensions of a known object
b) Calculate deformability of a known object
c) Determine force necessary to grasp known object by iteratively lifting object
d) Link object attributes and grasping force into a learning table

II. Retrieve the learned forces a robotic system must exert in order to grasp a deformable object on-line

a) Determine the 3D position and dimensions of an unknown object
b) Calculate the deformability of an unknown object
c) Map these object attributes into the learning table to retrieve grasping force

IV. Simulation Results

A simulation program was written which simulates the above learning process. In order to validate the process, Reznik and Laugier's algorithms are used to analytically model the deformation of the object as it is grasped.

Reznik and Laugier's Algorithm

A Particle System T is a tuple (P,S) made up of a set P of particles and a set S of springs. Each particle is represented by (**m**,**x**,**v**,**f**) where m is the particle's mass, **x** is the particle's position, **v** is the particle's velocity, and **f** is the force acting on the particle. Each spring is represented by (i,j,k,μ,l_0) where $p_i, p_j \in P$ are connected by the spring, k is the spring's elastic constant, μ is the damping factor, and l_0 is the rest length. Let $\mathbf{u}_{ij}$ be the force spring s applies to its endpoint particles p_i and p_j. Let $u_{ij} = [k(\|d\|-l_0) + \mu\|\dot{d}\|]\hat{d}$ where $k(\|d\|-l_0)\hat{d}$ is the resultant of the tensions of the springs linking p_i to its neighbors and $\mu\|\dot{d}\|\hat{d}$ is the viscous damping used to model in first approximation the dissipation of the mechanical energy of the model.

Given this initial model, we must now determine particle displacement as a known force is applied against the surface. In order to ensure spring convergence, we must calculate internal forces at an iterative instant of time. The period of an oscillatory particle system is given by $T_0 = \sqrt{\frac{m}{k}}$. The time factor must therefore by chosen such that $\Delta t \ll T_0$.

At each instance of time, the force acting on the particle consists of the summation of internal spring forces and any external force applied to the particle.

$$f_{\varphi_{ijk}} = \sum_{\forall ij \in \varphi_{ijk}} u_{ij} + \sum_{\forall j \in f_{applied}} f^{j}_{applied}$$

We can use the standard equations of motion to calculate all particle displacements at time t.

$$x(t + \Delta t) = x_0 + v\Delta t$$
$$v(t + \Delta t) = v_0 + a\Delta t$$
$$a(t + \Delta t) = \frac{f_{\varphi_{ijk}}(t)}{m}$$

Assumptions

We assume that:

1. Object is isotropic and homogeneous
2. Object displays perfectly linear elastic and time-independent behavior
3. At time t = 0, object is fixed in a stable position
4. Weight of the object is provided during run time.

Object Model

The object presented to the system is a six-sided convex object with each side possessing equivalent dimensions. After discretization, the object is represented as below (Fig. 4)

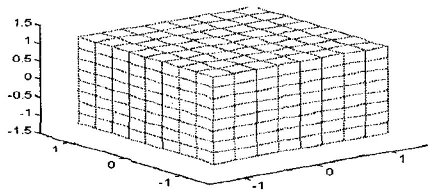

Fig. 4 Object Discretization

Technical Approach

To determine the forces necessary to lift a deformable object, we must first calculate the deformability of the object. Using the above described technique, we derive the graph for $f_{spring} = D(x)*x$ for objects with differing deformability factors. Based on the first two assumptions, this graph is linear in nature (Fig. 5).

Since we assume that the object exhibits time-independent behavior, the damping coefficient $\mu = 0$. Therefore, $f_{damped} = 0$ for all x, x'.

Although we have knowledge pertaining to the internal representation of the model, we experience some error in the retrieved value of D(x) due to the fact that, although we can visually retrieve the deformation distances along the outer edges of the object, we choose to only approximate the internal deformation in order to ensure calculations in real time (Fig. 5).

Once deformation is retrieved, we must now learn an adequate grasp. Using the above procedure, we can learn the necessary forces required for grasping. This learned force, along with weight and spring constant are then stored in a table. Since, the simulation only utilizes a small subset of objects, a simple look up table was sufficient for storage at this time.

After learning, an object is once again presented to the system. Based on the calculated deformation, the stored force is extracted and applied to the object for grasping.

V. Future Research

We next propose to implement the system in a real-world environment. The first step is to implement the system based on a list of assumptions similar to those as stated above. Upon successful completion of this task, we plan to relax the assumptions of homogeneity, isotropicity, and uniform elasticity to encompass the class of generic 3-D deformable objects.

VI. Conclusion

We have presented a generalized approach to handling of 3D deformable objects. Our task was to derive a new method for learning grasping characteristics for a non-rigid object represented by a physically-based model. Based on the results of the simulation, we determine that once we calculate the spring constant of the object, this factor, along with the given weight of the object, index into a database for extraction of force and displacement of the manipulator. To successfully lift the object, these two parameters are the necessary inputs required for the robotic system.

Bibliography

[1] Annaswamy, A.M. and Seto, D., "Object Manipulation Using Compliant Fingerpads: Modeling and Control", Transactions of the ASME, Vol. 115, Dec. 1993, pp. 638-648.

[2] Byars, E.F., and Snyder, R.D., Engineering Mechanics of Deformable Bodies, International Textbook Company, Penn., 1969.

[3] Hopcroft, J., Kearney,J., and Krafft, D., "A Case Study of Flexible Object Manipulation", Int. Journal of Rob. Research, Vol. 10, No. 1, Feb. 1991, pp. 41-50.

[4] Logan,I.P., Wills, D.P.M., and Avis N.J., "Deformable Objects in Virtual Environments", Sec. UK VR-SIG Conference, U.K. Dec. 1994.

[5] Meer,D.W., and Rock, S.M., "Experiments in Object Impedance Control for Flexible Objects", Proc. Int. Conf. on Robotics and Automation, May 1994.

[6] Montana, D.J., "The Kinematics of Contact with Compliance", Proc. Int. Conf. on Robotics and Automation, 1989, pp. 770-774.

[7] Patton, R., Swern, F., Tricamo, S., and van der Veen, A., "Automated Cloth Handling Using Adaptive Force Feedback", Jour. of Dynamic Systems Measurements and Control, Vol.114, Dec. 1992, pp. 731-733.

[8] Provot, Xavier, "Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior"

[9] Reznik, D., and Laugier, C., "Dynamic Simulation and Virtual Control of a Deformable Fingertip", submitted to ICRA '96 conference, 1996.

[10] Sinha, P.R., Xu, Y., Bajcsy, R.K., and Paul, R.P., "Robotic Exploration of Surfaces with a Compliant Wrist Sensor", International Journal of Robotics Research, Vol. 12, No. 2, April 1993.

[11] Shimoga, K.B., and Goldenberg,A.A., "Soft Robotic Fingertips Part II: Modeling and Impedance Regulation", Int. Journal of Rob. Research, Vol 15, No. 4, Aug 1996, pp. 335-350.

[12] Terzopoulos, D., Platt, J., Barr, A., and Fleischer, K., "Elastically Deformable Models", Computer Graphics, Vol. 21, No. 4, July 1987, pp. 205-214.

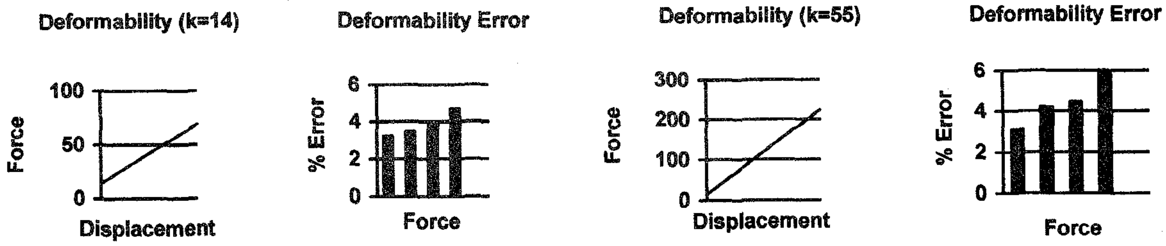

Fig. 5 Sample of Learned Deformability Constant and Error Graph

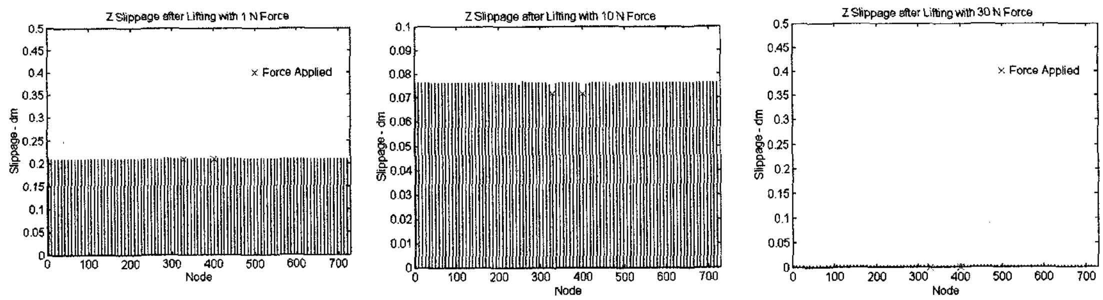

Fig. 6 Sample of Grasping Force vs. Slippage a) 1 N Force, b) 10 N Force, c) 30 N Force. 30N Force is Grasping Force with no slippage. Graph c is sample of Learned Grasping Force.