Non Contact 2D and 3D Shape Recognition by Vision System for Robotic Prehension

Bikash Bepari, Ranjit Ray and Subhasis Bhaumik
Haldia Institute of Technology, Haldia, India
Central Mechanical Engineering Research Institute, Durgapur, India
Bengal Engineering and Science University, Shibpur, Howrah, India

1. Introduction

The increasing demand for robotic applications in today's manufacturing environment motivates the scientists towards the design of dexterous end-effectors, which can cope with a wide variety of tasks. The human hand can serve as a model for a robotic hand for prehension, which interacts with the environment with more than twenty-seven degrees of freedom (dof^s). The articulation and accommodation of the human hand in terms of dexterity, stability, tactile and /or nontactile sensation, stiffness, weight to grip force ratio, resistance to slip, and adaptability provide clear direction for active research in dexterous manipulation by the robot.

Robotic hands, which have been developed over past three decades, sometimes on adhoc basis to accomplish a predefined task. In robotic prehension, perhaps the choice of grip to ensure stable grasp has become the main hurdles to confront with. Since the inception of development of robotic hand, this has been revealed from thorough research that, more emphasis has been given on the electro-mechanical considerations than on the stability of the grasp, that too under vision assistance. This leads to an effort to develope new algorithms for stable grasp irrespective of its shape.

Prehension combines the choice of grip and the act of grasping including its control. A desire, internally or externally generated, triggers responses in the memory and the eyes; the hand instantaneously takes a position over the object, grips it and manipulates the task. The process of prehension is controlled by feedback loops with the position and the force of the fingers as shown in Fig.1. The eye is basically a sensor, whereas the brain is the central processing unit, which sends signal to the different muscles and tendons to act accordingly to actuate the hands for manipulation. Choosing a grip is an important step in the process of prehension. The block diagram in Fig. 2 illustrates the relationship between the different activities included in prehension. This can be simulated by using a vision system on a robot and a processor with feedback loop to manipulate the tasks by an appropriate gripper in the similar pattern of eye-memory analysis as shown in Fig. 3.

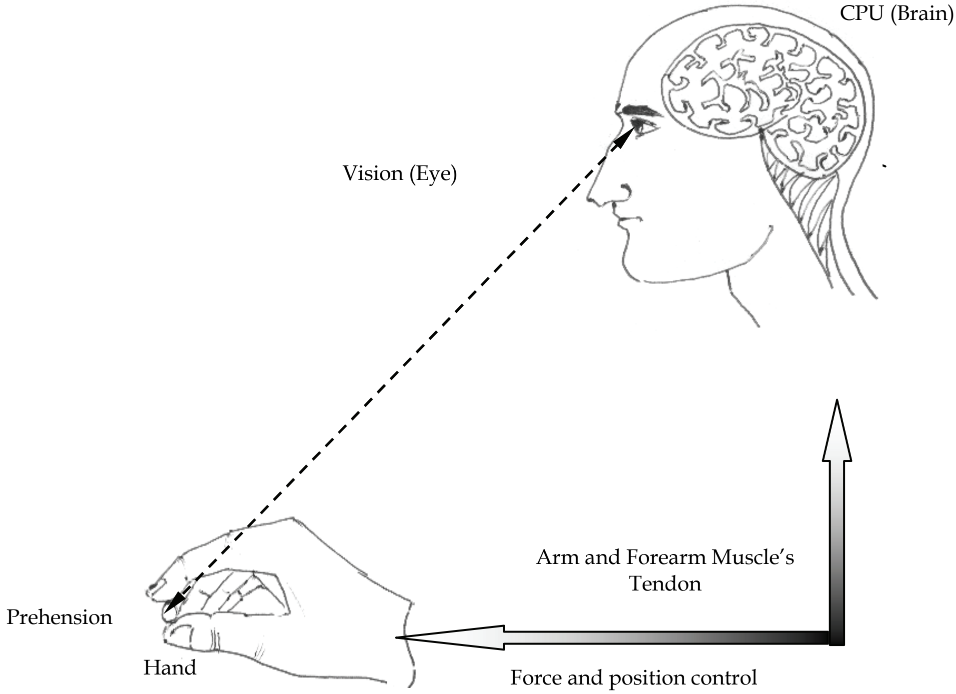

Fig. 1. The eye -memory integration-based human prehension

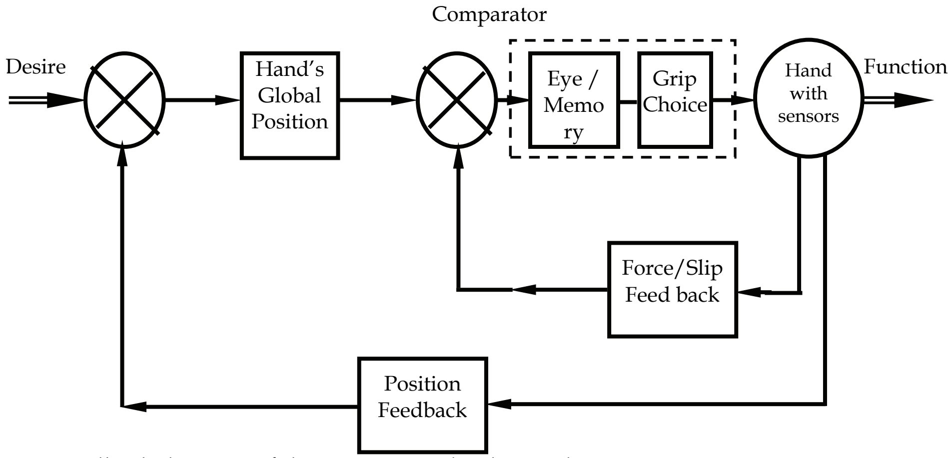

Fig. 2. Feedback diagram of the actions involved in prehension

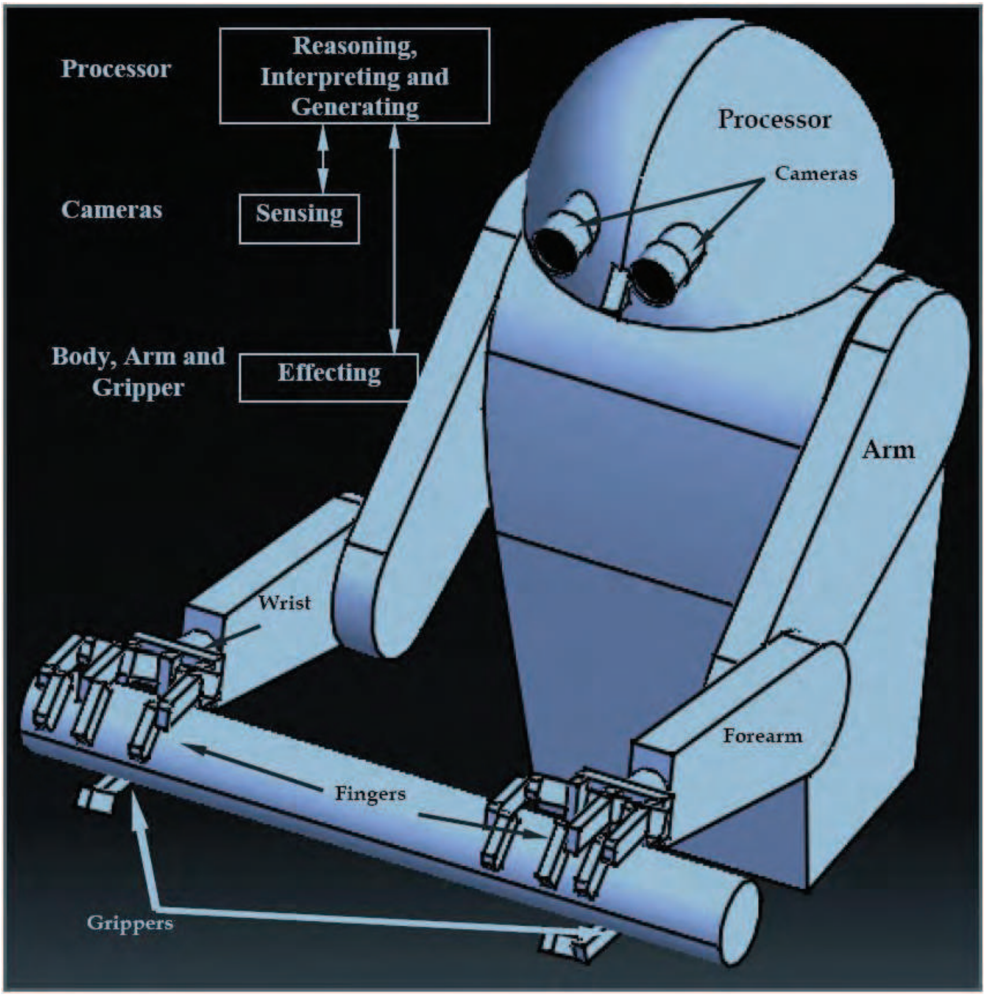

Fig. 3. Simulation of the robotic prehension

The concept of replication of the attributes of human morphology, sensory system and neurological apparatus alongwith the behavior leads to a notion of embodiment – this in turn over time is refined, as the brain and physiology change. If the grasping modes are critically examined between a child and an adult, a distinguishable difference may be observed between the two. The presense of simultaneous path planning and preshaping differentiates the latter one from the former.

The essential goal of the present work was to ensure the stability of a grasp under visual guidance from a set of self generating alternatives by means of logical ability to impart adroitness to the system (robotic hand).

2. Survey of Previous Research

As a step towards prehension through vision assistance in robotic system, Geisler (Geisler, 1982) described a vision system consisting of a TV camera, digital image storage and a mini computer for shape and position recognition of industrial parts.

Marr (Marr, 1982) described 3-D vision as a 3-D object reconstruction task. The description of the 3D shape is to be generated in a co-ordinate system independent of the viewer. He ensures that the complexity of the 3-D vision task dictates a sequence of steps refining descriptions of the geometry of the visible surfaces. The requirements are to find out the pixels of the image, then to move from pixels to surface delineation, then to surface orientation and finally to a full 3-D description.

Faugeras (Faugeras, 1993) established a simple technique for single camera calibration from a known scene. A set of '**n**' non-co-planar points lies in the 3-D world and the corresponding 2-D image points are known. The correspondence between a 3-D scene and 2-D image point provides an equation. The solution so obtained, solves an over-determined system of linear equations. But the main disadvantage of this approach is that the scene must be known, for which special calibration objects are often used. Camera calibration can also be done from an unknown scene. At least two views are needed, and it is assumed that the intrinsic parameters of the camera do not change.

Different researchers like (Horaud et al., 1995), (Hartley, 1994) and (Pajdha & Hlavac, 1999) worked on this approach. Horaud considered both rotational and translational motion of the camera from one view to another. Hartley restricted the camera motion to pure rotation and Pajdha et al. used pure translational motion of camera to get linear solution. Sonka et al. (Sonka, 1998) discussed on the basic principle of stereo vision (with lateral camera model) consisting of three steps: a) Camera calibration, b) Establishing point correspondence between pairs of points from the left and the right image and c) Reconstruction of 3D coordinates of the points in the scene.

David Nitzan (Nitzan, 1988) used a suitable technique to obtain the range details for use in robot vision. The methodology followed in his work is, image formation, matching, camera calibration and determination of range or depth. Identifying the corresponding points in two images that are projections of the same entity is the key problem in 3D vision. There are different matching techniques for finding the corresponding points.

Victor et al. (Victor & Gunasekaran, 1993) used correlation formula in their work on stereovision technique. They determined the three-dimensional position of an object. Using the correlation formula they have computed the distance of a point on an object from camera and have shown that the computed distance from camera is almost equal to the actual distance from camera.

Lee et al. (Lee et al., 1994) used perspective transformation procedure for mapping a 3-D scene onto an image plane. It has also been shown that the missing depth information can be obtained by using stereoscopic imaging techniques. They derived an equation for finding the depth information using 3-D geometry. The most difficult task in using the equation for obtaining the depth information is to actually find two corresponding points in different images of the same scene. Since, these points are generally in the same vicinity, a frequently used approach is to select a point within a small region in one of the image views and then attempt to find the best matching region in the other view by using correlation techniques. The geometry of a 3-D scene can be found if it is known which point from one image corresponds to a point in the second image. This correspondence problem can be reduced by using several constraints.

Klette et al. (Klette et al., 1996) proposed a list of constraints like uniqueness constraint, photometric compatibility constraint, and geometric similarity constraint etc that are commonly used to provide insight into the correspondence problem. They illustrated this approach with a simple algorithm called block matching. The basic idea of this algorithm is that all pixels in the window (called a block) have the same disparity, meaning that one and only one disparity is computed for each block.

But later on Nishihara (Nishihara, 1984)noted that such point-wise correlators are very heavy on processing time in arriving at a correspondence. He proposed another relevant approach to match large patches at a large scale, and then refine the quality of the match by reducing the scale using the coarser information to initialize the finger-grained match.

Pollard et al. (Pollard et al., 1981) developed the PMF algorithm using the feature-based correspondence method. This method use points or set of points those are striking and easy to find, such as pixels on edges, lines or corners. They proceed by assuming that a set of feature points [detected edges] has been extracted from each image by some internal operator. The output is a correspondence between pairs of such points.

Tarabanis et al. (Tarabanis & Tsai, 1991) described the next view planning method as follows: "Given the information about the environment as well as the information about that the vision system has to accomplish (i.e. detection of certain object features, object recognition, scene reconstruction, object manipulation), develop strategies to automatically determine sensor parameter values that achieve this task with a certain degree of satisfaction".

Maver et al. (Maver & Bajcsy, 1993) proposed an NVP (Next View Planning) algorithm for an acquisition system consisting of a light stripe range scanner and a turntable. They represent the unseen portions of the viewing volume as 2½-D polygons. The polygon boundaries are used to determine the visibility of unseen portions from all the next views. The view, which can see the largest area unseen up to that point, is selected as the next best view.

Connolly (Connolly, 1985) used an octree to represent the viewing volume. An octree node close to the scanned surface was labeled to be seen, a node between the sensor and this surface as empty and the remaining nodes as unseen. Next best view was chosen from a sphere surrounding the object.

Szeliski (Szeliski, 1993) first created a low-resolution octree model quickly and then refined this model iteratively, by intersecting each new silhouette with the already existing model. Niem (Niem, 1994) uses pillar-like volume elements instead of an octree for the model representation.

Whaite et al. (Whaite & Ferrie, 1994) used the range data sensed to build a parametric approximate model of the object. But this approach does not check for occlusions and does not work well with complex objects because of limitations of a parametric model.

Pito (Pito, 1999) used a range scanner, which moves on a cylindrical path around the object. The next best view is chosen as the position of the scanner, which samples as many void patches as possible while resampling at least a certain amount of the current model.

Liska (Liska, 1999) used a system consisting of two lasers projecting a plane onto the viewing volume and a turntable. The next position of the turntable is computed based on information from the current and the preceding scan.

Sablating et al. (Sablating et al., 2003; Lacquaniti & Caminiti, 1998). described the basic shape from Silhouette method used to perform the 3-D model reconstruction. They experimented with both synthetic and real data.

Lacquaniti et al. (Lacquaniti & Caminiti, 1998) reviewed anatomical and neurophysical data processing of a human in eye-memory during grasping. They also established the different mapping techniques for ocular and arm co-ordination in a common reference plane.

Desai (Desai, 1998) in his thesis addressed the problem of motion planning for cooperative robotic systems. They solved the dynamic motion-planning problem for a system of cooperating robots in the presence of geometric and kinematic constraints with the aid of eye memory co ordination.

Metta et al. (Metta & Fitzpatrick, 2002) highlighted the sensory representations used by the brain during reaching, grasping and object recognition. According to them a robot can familiarize itself with the objects in its environment by acting upon them. They developed an environment that allows for a very natural developmental of visual competence for eye-memory prehension.

Barnesand et al. (Barnesand & Liu, 2004)developed a philosophical and psycho-physiological basis for embodied perception and a framework for conceptual embodiment of vision-guided robots. They argued that categorization is important in all stages of robot vision. Further, classical computer vision is not suitable for this categorization; however, through conceptual embodiment active perception can be erected.

Kragic et al. (Kragic & Christensen, 2003) considered typical manipulation tasks in terms of a service robot framework. Given a task at hand, such as “pick up the cup from the dinner table”, they presented a number of different visual systems required to accomplish the task. A standard robot platform with a PUMA560 on the top is used for experimental evaluation.

The classical approach-align-grasp idea was used to design a manipulation system (Bhaumik et al., 2003). Both visual and tactile feedback was used to accomplish the given task. In terms of image processing, they started by a recognition system, which provides a 2-D estimate of the object position in the image. Thereafter, a 2-D tracking system was presented and used to maintain the object in the field of view during an approach stage. For the alignment stage, two systems are available. The first is a model based tracking system that estimates the complete pose/velocity of the object. The second system was based on corner matching and estimates homography (matching of periphery) between two images. In terms of tactile feedback, they presented a grasping system that performs power grasps. The main objective was to compensate for minor errors in object's position/orientation estimate caused by the vision system.

Nakabo et al. (Nakabo et al., 2002) considered real-world applications of robot control with visual servoing; both 3-D information and a high feedback rate is required. They developed a 3-D target tracking system with two high-speed vision systems called Column Parallel Vision (CPV) systems. To obtain 3-D information, such as position, orientation and shape parameters of the target object, a feature-based algorithm has been introduced using moment feature values extracted from vision systems for a spherical object model.

3. Objective

In the present investigation, an attempt has been made to enable a four-fingered robotic hand consisting of the index finger, middle finger, ring finger and the thumb to ensure stable grasp. The coordinated movement of the fingertips were thoroughly analyzed to preshape the fingers during trajectory planning in order to reduce task execution time. Since the displacement of the motor was coordinated with the motion of the fingertips, the corelation between these two parameters was readily available thorugh CAD simulation using Visual Nastran 4D (MSC. VisualNastrun 4D, 2001).

The primary objectives of the present investigation are:

a) analysis of the object shapes and dimensions using 2D image processing techniques and vision based preshaping of the finger's pose depending on the basis of prehension,

b) hierarchical control strategies under vision guidance for slip feedback,

c) experimentation on the hand for intelligent grasping,

4. Brief Description of the Setup for Prehension

4.1 Kinematics of the Hand

The newly developed hand uses a direct linkage mechanism to transfer motions. From Fig.4 it is clear that the crank is given the prime motion, which ultimately presses the push link-1 to turn the middle link about the proximal joint. As the middle link starts rotating, it turns the distal link simultaneously, because the distal link, middle link and push link-2 form a crossed four bar linkage mechanism. The simultaneous movement of the links ultimately stops when the lock pin comes in contact with the proximal link. The proximal link is kept in contact with the stay under the action of a torsional spring. As the lock pin comes in contact with the proximal link, it ultimately restricts all the relative motions between the links of the finger and at that position the whole finger moves as an integrated part. The crank is coupled to a small axle on which a worm wheel is mounted and the worm is directly coupled to a motor by coupling as shown in Fig.5.

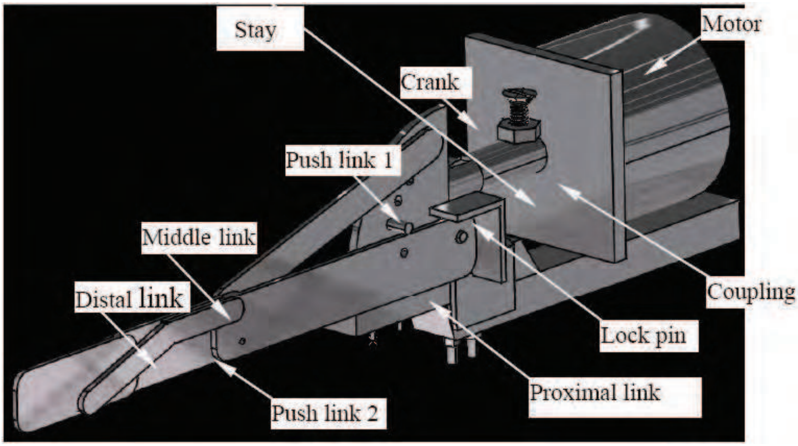

Fig. 4. The kinematic linkages for the robot hand

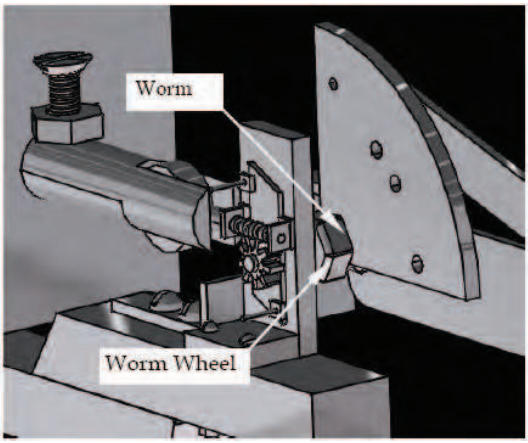

Fig. 5. The worm worm-wheel mechanism for actuation of the fingers

4.2 Description of the Mechanical System

The robotic hand consists of four fingers namely thumb, index, middle and ring fingers. Besides, there is a palm to accommodate the fingers alongwith the individual drive systems for actuation. The base, column and the swiveling arm were constructed in this system for supporting the hand to perform the experiments on grasping. The design of the base, column and swiveling arm were evolved from the absence of a suitable robot.

The base is mild steel flat on which the column is mounted. On the free end of the column, a provision (hinge) has been made to accommodate the swiveling arm adaptor. The CAD model of the setup has been shown in Fig.6.

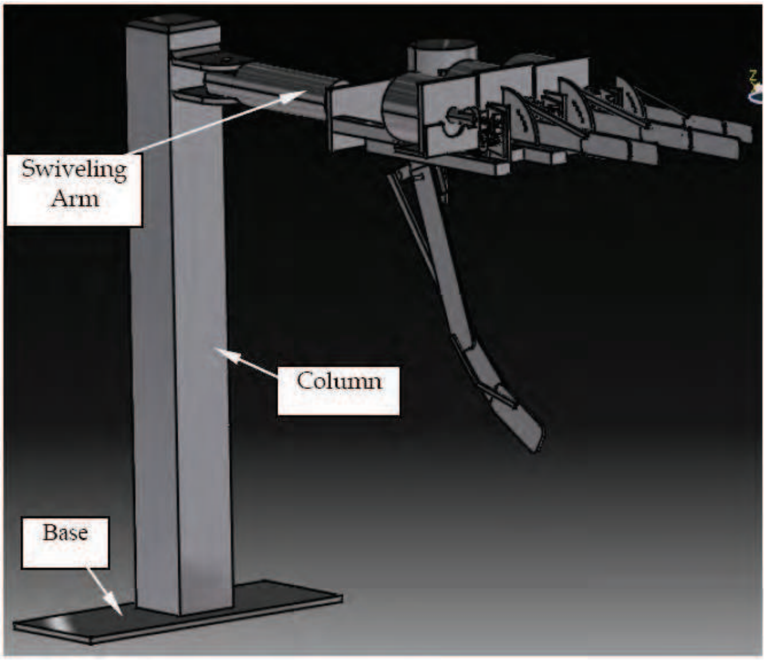

Fig. 6. The assembled CAD Model of the robot hand

4.3 Vision System

The specification of the camera is as follows:

a.	Effective picture element	:	752 (H) × 582 (V)
b.	Horizontal frequency	:	15.625 KHz ± 1%
c.	Vertical Frequency	:	50Hz ± 1%
d.	Power requirements	:	12v DC ± 1%
e.	Dimension	:	44(W) × 29 (H) × 57.5(D) mm
f.	Weight	:	110 gm

A two-channel PC based image-grabbing card was used to acquire the image data through the camera. Sherlock ™ (Sherlock) is a window based machine vision environment specifically intended to simplify development and deployment of high performance alignment, gauging inspection, assembly verification, and machine guidance tasks. This was used to detect all the peripheral pixels of the object being grasped after thresholding and calibration. The dimensional attributes are solely dependant on the calibration. After calibration a database was made for all the peripheral pixels. The camera alongwith the mounting device has been shown in Fig.7.

Fig. 7. The Camera alongwith the mounting device

5. Fingertip Trajectories and Acquisition of Pre-shape Values

To determine the trajectories of the fingers in space, points on the distal palmer tip were chosen for each of the fingers and during simulation the locus of those points were traced. The instantaneous crank positions were also taken simultaneously. As the fingers flex, the coordinate abscissa (X) value either increases or decreases depending on the position of the origin as the current system. Since the incremental values for coordinate (X) are correlated with the angular movement of the crank, during preshape of the fingers, proper actuation can be made as shown in Fig.8. Once the vision based sensory feedback values are known, the motors may be actuated to perform the required amount of incremental movements.

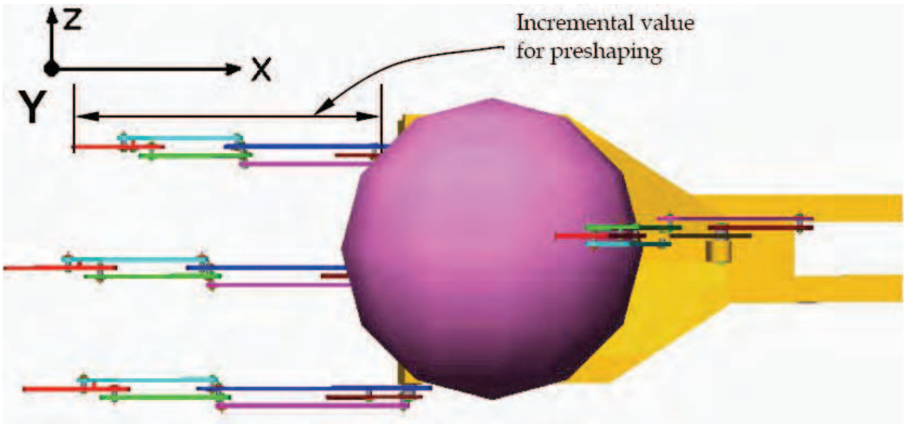

Fig. 8. The preshape value for the fingers

The model was so made that the direction of the finger axis was the X-axis and the gravity direction was the Y direction. Figure 9 shows the trajectories for different fingers. The correlations of the incremental values in preshaping direction and the corresponding crank movement have been shown in Fig.10, Fig.11, Fig.12 and Fig.13. The R^2 values in the curves imply the correlation constant and as the value tends to 1 (one) implies a good correlation of the curve fitting.

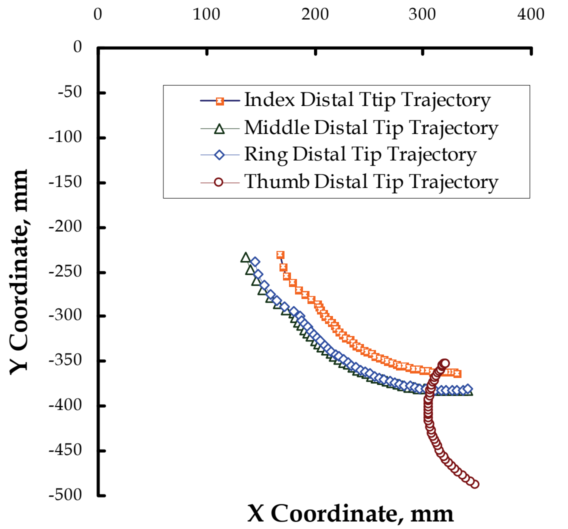

Fig. 9. The trajectories of the fingertips of the fingers

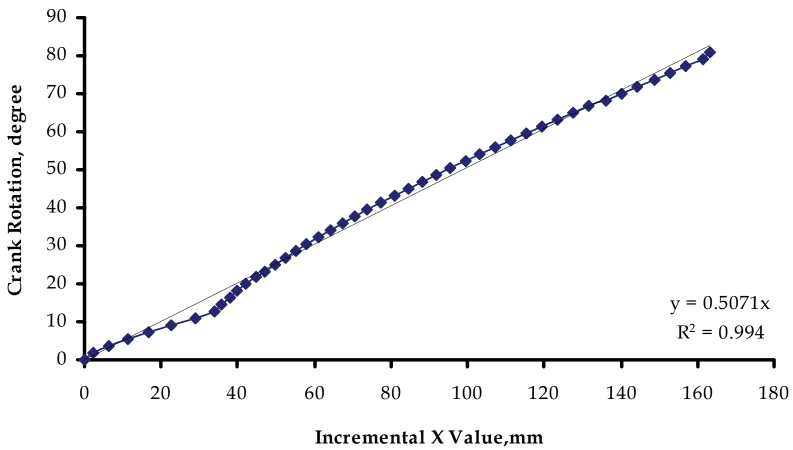

Fig. 10. Preshape value vs. crank rotation angle plot for index finger

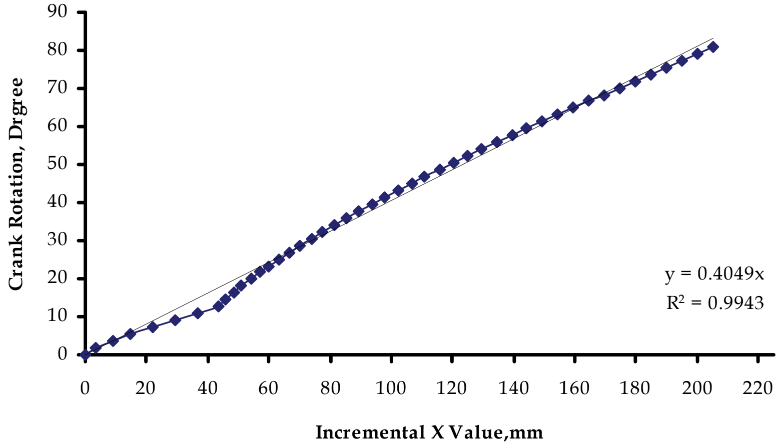

Fig. 11. Preshape value vs. crank rotation angle plot for middle finger

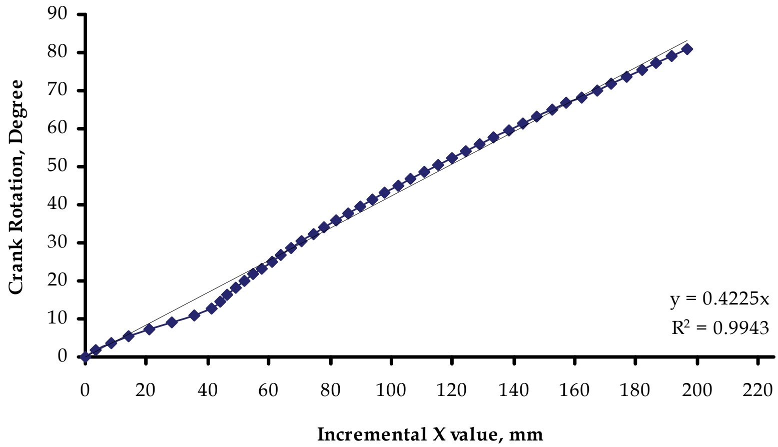

Fig. 12. Preshape value vs. crank rotation angle plot for ring finger

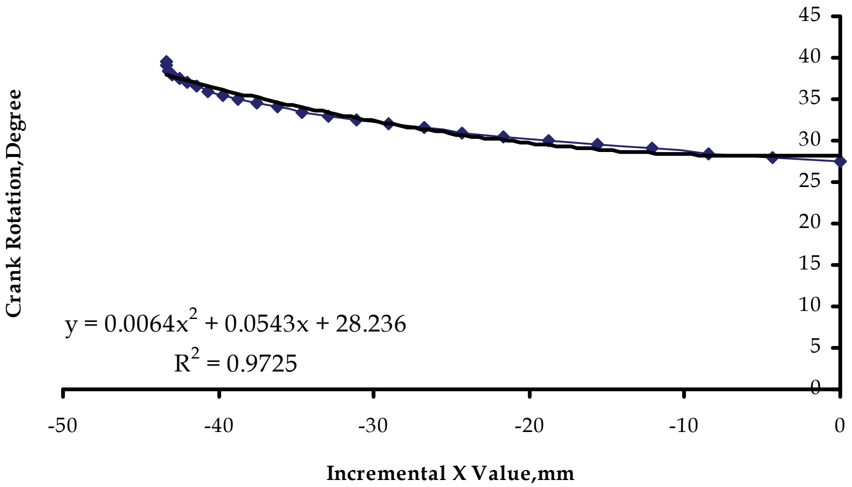

Fig. 13. Plot showing preshape value vs. crank rotation angle for thumb

6. Control Hierarchy and Vision Assistance

The control architecture is a three level hierarchy, which consists of the following blocks:

6.1 A computer with CCD Camera i.e. Master Computer–I

The computer at the top level of hierarchy gives the object shape information to the next level computer to actuate the motors for preshaping through the slave i.e. the microcontroller motherboard.

6.2 The Master Computer-II

The second level computer acting in master mode communicates with the micro-controller motherboard that acts as slave through the serial port, RS 232.

6.3 Controller Mother Board with Micro Controller Chip (AT89C52)

The main function of micro-controller motherboard is to behave like that of a Data Acquisition System (DAS) and thus helps the master by providing the required data from different sensors. Evaluating the controller mode equation through software, the master downloads the firing angle information and motor direction information in to the micro-controller chip (AT89C52), which in turn sends these signals to the drive circuitry. Figure 14 represents the realistic view for the three-generation controller. Figure 15 depicts the overall control architecture used in this scheme.

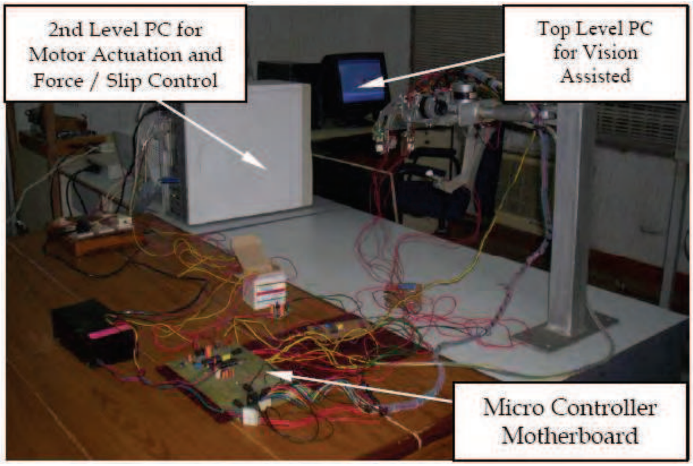

Fig. 14. Prehension system with three level hierarchy

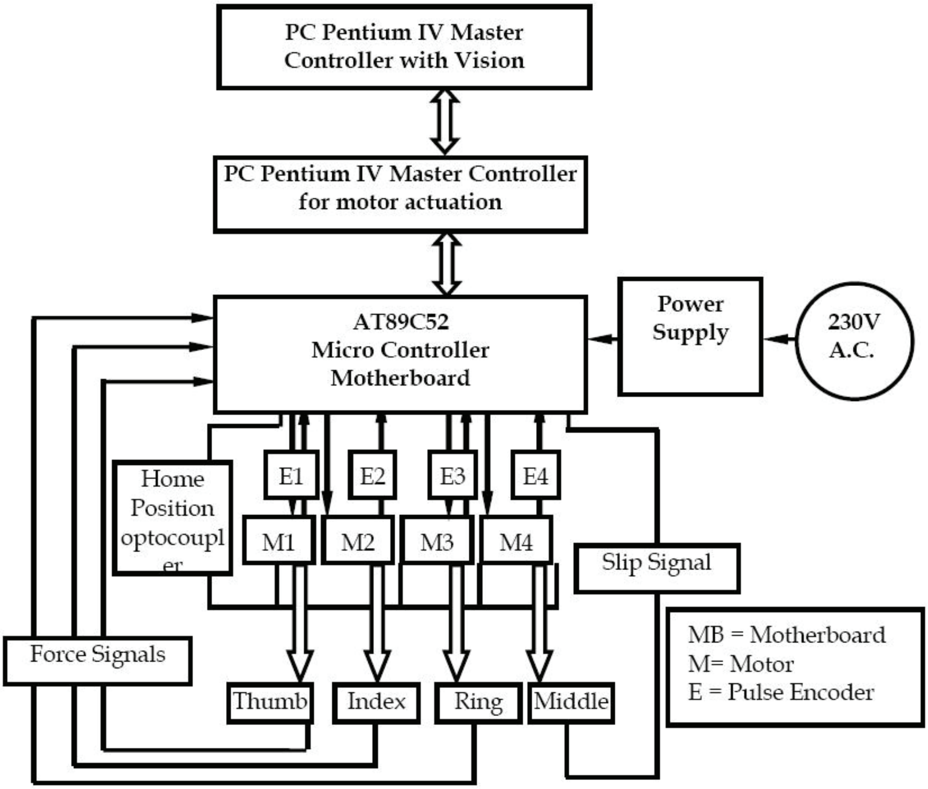

Fig. 15. The three level hierarchy of control system

6.4 Actuation of Motors under Vision Assistance

The motors are acting as actuators for finger motion as the finger positions are determined by the motor position. The vision assistance helps the fingers to preshape to a particular distance so that the fingertips are a few units apart from the object. As soon as the fingers touch the object, the force sensor reads the current value of the force exerted by all the fingers. The incremental step is given to all the motors as 7.2° i.e. four counts.

The control strategy is divided into three steps:

a) The object shape information can be had by CCD camera to generate the amount of preshape (i.e. the pulse count by which the motor will rotate to encompass the job). The initial set value for the position comes from the vision assistance. The scheme has been depicted in Fig.16.

b) Secondly it is to generate a database for grasping force exerted by individual finger for grasping different materials under no slip condition. The amount of slip is then fed back to the computer and the computer through microcontroller sends signal to the motors via motor drive card to move forward. The total control strategy to generate the database of force at no slip condition has been shown in Fig.16.

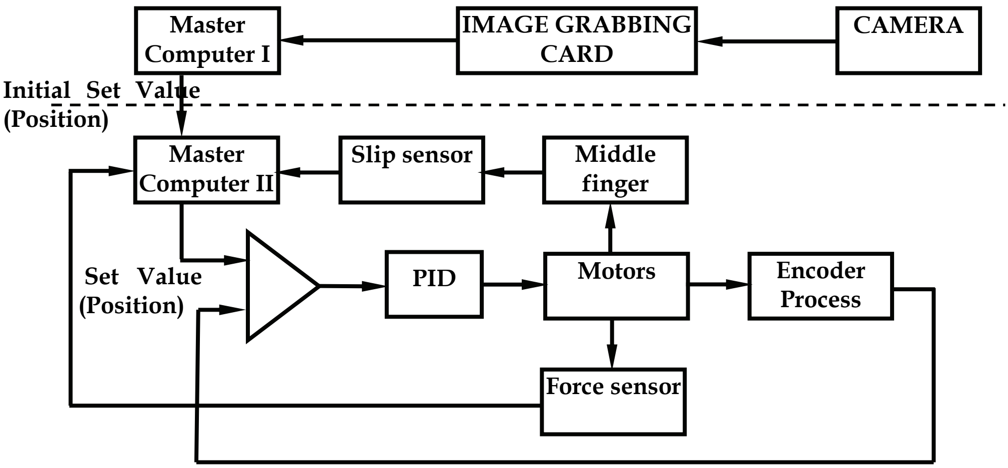

Fig. 16. The vision based preshaping and slip feedback

Then the values of the grasping forces at which slip stops is feed to the controller i.e. the grasping force exerted by the individual fingers for different objects. Once the set value is reached then the controller simultaneously checks whether there is any slip or not. If till now the fingers are not capable to grip, then the set values are dynamically increased to a maximum of 10% of the initial set values with an increment of 1%. Beyond which the system will declare that the process is out of range.

This process approaches towards adaptive control. The schematic diagram of force control has been shown in Fig.17. The microcontroller motherboard is connected to the computer by RS-232 serial port. But position forward scheme does not imply that there will always be a force signal, rather the fingertips are to be in contact with the object at first, only then the force signals can be achieved.

In order to control the grasp position of the gripper for holding an object securely the motor should achieve the desired set position smoothly without any overshoot and undershoot obviously with negligible offset error. This set position corresponds to the stable grasp position. To control the gripping force required for an object, a PID closed loop feedback control scheme is adopted. The initial set value for position comes from the vision data and the motors rotate to preshape the objects dimension and to see whether the gripper is in contact with the object or not. If the gripper is not in contact with the object then the motor moves forward till all the fingers come in contact with the job. If the contact exists, then the PID controller sets the current process value plus four counts as set value and checks whether there is any slip or not. Now if there is a slip, the set value is increased by another four counts until no slip condition is reached and the master computer–II saves all the optimum grasping value in the database. The flow chart has been depicted in Fig.18.

In the second scheme i.e. the force control scheme, the initial set values for force comes from the database of the master computer. Feed forward signal are given to the motors for wrapping the object. This process continues till the desired set values for all the fingers are reached. Then it is ensured whether there is any slip or not after reaching to the set position. If till now there is any slip then the set value will be set as process value plus 1% of the set value until slippage stops and gripper holds the object securely. Each fingertip contains electrical strain gauge type force sensing elements for the measurement of the forces. The control flowchart scheme has been shown in Fig.19.

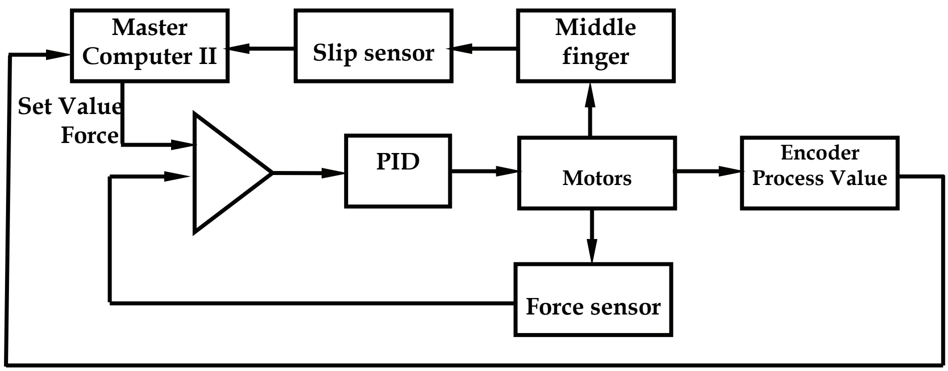

Fig. 17. The force feedback for stable grasp

The main advantage of this composite control, PID mode is that the integral mode eliminates the offset error produced by proportional mode and the overshoot and undershoot of the controlled output is minimized by derivative action through its anticipatory property. The overall effect is that the output-controlled variable achieves the desired value without much cycling. Critically damping condition has been implemented in the present work. The control software has got the feature to set the set value manually or automatically (from the database). The gain values can also be set from the software as shown in Fig.20.

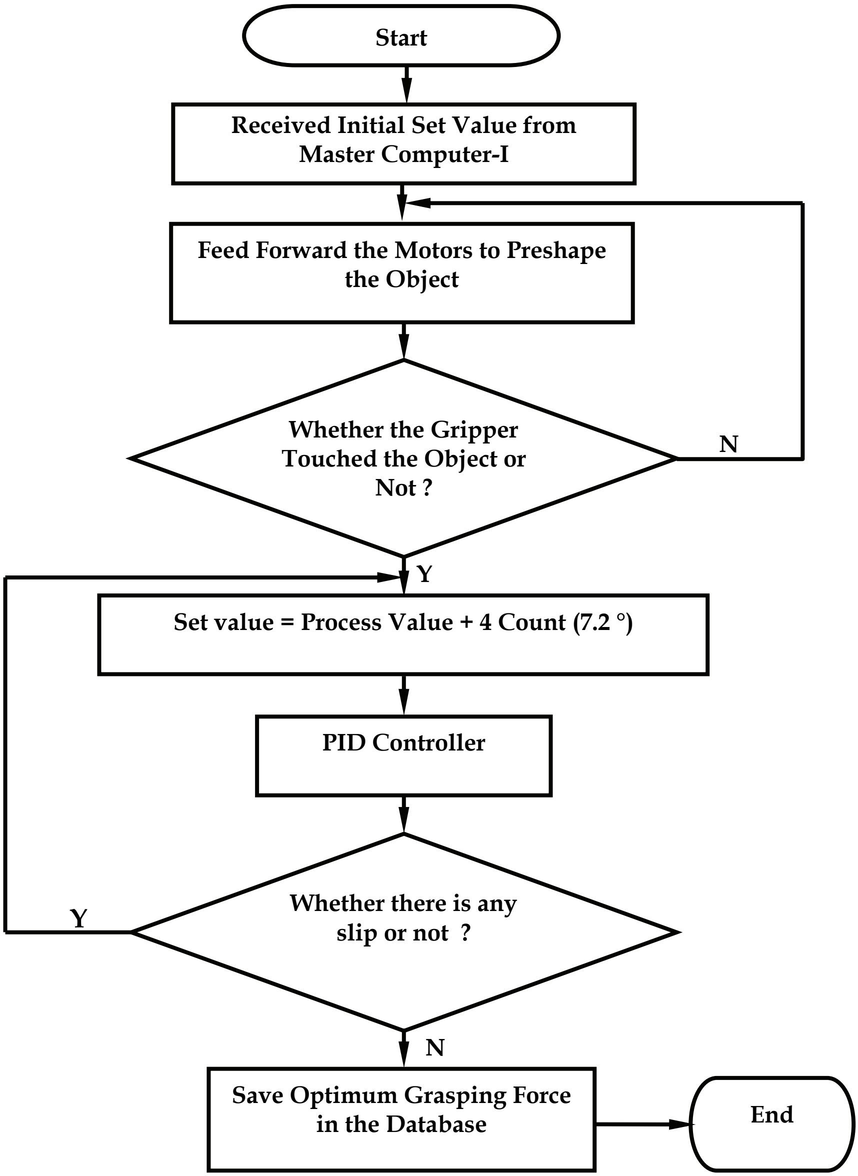

Fig. 18. The flowchart depicting vision assisted preshape and slip feedback