Assistive Feeding System: Design and Evaluation

Usama Jahangir*, Wajid Ali, Muhammad Fahad Aamir, Mohsin Islam Tiwana, Hamid Jabbar
Department of Mechatronics Engineering, College of Electrical and Mechanical Engineering, National University of Sciences and Technology, Islamabad, Pakistan
*email: mr.usama.jahangir@gmail.com

***Abstract*— Old people and patients with limited or complete motor impairment had always face difficulty in their daily life task in which most important and frequent is eating meals, such people need assistant like nurse to feed them. In this paper we worked on an assistive feeding system based on 6 Degree of Freedom fixed base open-source robotic arm with active assistance. The manipulator is 3D printed by fused deposition modelling technique using PLA+ material and a counterweight mechanism is introduced to reduce the motor torque requirements hence reducing the power consumption of the system. The system is divided into 2 sub-systems which are hybrid-vision system to get location of mouth of the person to feed and, scooping and delivery system which uses manipulator and control system to feed the food. The system is completely embedded on raspberry pi with wireless access for user and programmer. Finally, we discuss the results of our system and limitations in the methodologies used.**

Keywords—Serial manipulators, BC-N3D Moveo, Kinematics, 3D localization, Counter weight mechanism

I. INTRODUCTION

According to a report by the World Health Organization (WHO) in 2011, approximately 15% of the world's population lives with some form of disability [1]. In another report by WHO, approximately 2.5 billion people are in need of some kind of assistive products which are projected to be around 15 billion by 2050 [2]. These numbers are alarming, and are required to be addressed.

Among aforementioned, people with upper limb disabilities, patients with contagious diseases and weak elderly face difficulty in doing basic daily tasks among which one of the most important is eating meals. Such people require assistance which is generally provided by a human care giver. But according to a report by the International Council of Nurses (ICN) 2021 there will be a shortage of 13 million nursing staff [3]. One solution to help such people is the use of assistive products to feed them food which is also considered by researchers at different times [4,5,6]. Several serial and parallel robotics manipulators are being used in the commercial and research sector to perform this task.

Although there are assistive robots that are commercially available including but not limited to My Spoon [7], Mealtime partner [8], and Meal budy [9], these robots provide limited assistance. In [10] such robots are referred to be providing *passive feeding* assistance, where the feeding position is predefined and the person has to use his upper body to take the food. On the contrary, *active feeding* assistance is based on up-to-date position of mouth. In [11] critical design approaches, feeding methods and advantages and limitations of different control methods of assistive feeding devices are discussed.

In this paper, we have utilized culmination of hybrid techniques to feed food. The content of this paper includes following;

- We present a counter weight mechanism to reduce torque requirement of serial manipulator ultimately reducing the cost with down side of reduce work envelope
- We present hybrid kinematics solution, utilizing combination of closed form and numerical solutions to solve inverse kinematics
- We present hybrid 3D localization technique, using sensor fusion of monocular camera and laser depth sensor to get target pose matrix

Such a system could significantly benefit healthcare infrastructure by offering autonomous, independent care for patients with contagious diseases, elderly individuals in home or assisted living settings, and people with disabilities.

II. SYSTEM CONFIGURATION

A. System Overview

The system is based on a modified open source 6 DoF (degrees of freedom) robotic arm; moveo by BCN3D [12], a monocular camera mounted on the end-effector, a laser sensor (VL53L0X), modified end-effector for holding utensils, counter weight mechanism and raspberry pi based control system. The experimental setup is such that the serial manipulator is placed on a wooden board acting as a fixed base, which also includes a bowl at one edge for food and the control circuit on the opposite edge. We assume that the location of food is predefined.

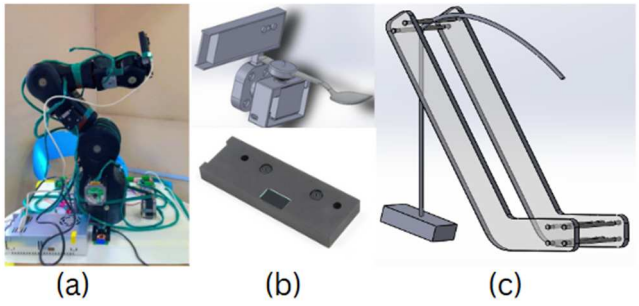

Fig 1: a) BCN3D Moveo, b) Modified end-effector on top and monocular camera on bottom, c) Counter weight mechanism

The system can perform three sub tasks: calibration, scooping and feeding. At first initialization, the system calibrates soft home with respect to hard home based on limit switches, afterwards user can recalibrate the system as per requirement. For scooping, robot moves to a predefined position to scoop food. Finally in feeding the robot moves to a predefined position or at safe distance from person's mouth to feed depending upon current mode i.e. passive or active. In case of active feeding an image will be acquired and processed to detect presence of face in field of view. If face is detected, localization is done using monocular depth estimation using facial landmarks leading to target transformation matrix calculation. Once target matrix is generated, inverse kinematic solution is calculated and robot is actuated to achieve the required pose matrix, at this stage laser sensor data is monitored to avoid collision ensuring safety. At last, feeding is performed and cycle is completed.

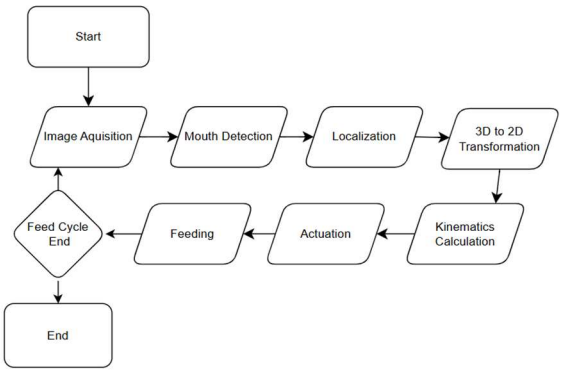

Fig 2: Assistive Feeding System overview

B. *Operating Procedure*

In this section, we discuss operating flow of the system when a person wants to eat food (e.g. Rice). We assume that person is inside the work envelop of the robot i.e. within 1 m range. Moreover, we assume that either user is able to use GUI using a tablet/mobile device or a second person is available remotely to control the robot. Fig 3 shows a single complete feeding cycle.

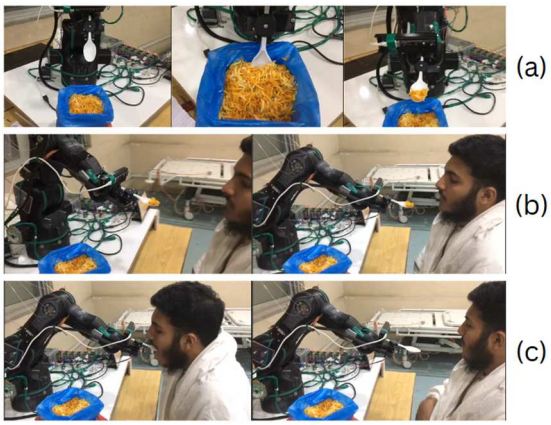

Fig 3: Single feeding cycle a) scooping b) localization c) feeding

When the scooping command is sent, the robot scoops food from predefined location. After that it waits for the user to prompt for feeding, depending on current mode either the robot takes the spoon at a predefined position or detects the person's face along with locating the mouth and stops a safe distance from mouth to avoid hitting.

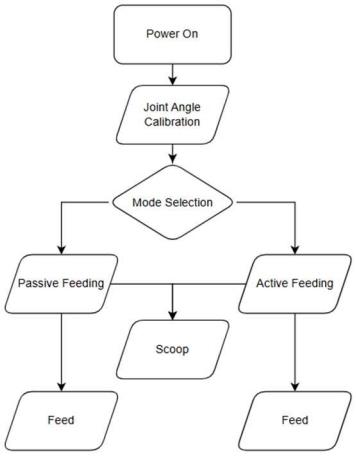

Fig 4: Operational flow of feeding system

III. METHADOLOGY

This research study is based on experimental results including the physical limitation of hardware. The methodology incorporates customized open-source robotic arm manufactured using fused deposition modeling (FDM) [13], Hybrid approach in formulating the kinematics using both the closed form solution and numerical solution and hybrid method for 3D localization using monocular depth estimation and sensor fusion with laser depth sensor.

A. *Mechanical Design Modifications*

The robotic arm used in the research is a modified version of open-source robot BCN3D Moveo. The original gripper from robot was replaced with a custom end-effector designed based on requirements of our feeding system including mounting of a camera, a laser sensor, a servo motor and switchable utensils holder with can be seen in fig 5.

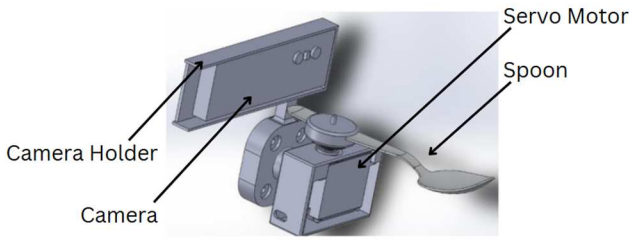

Fig 5: a) Custom end-effector CAD model

During initial experiments, the robot had a torque shortage due to use of low torque stepper motors due to shortage of required actuator in the Pakistani market at the time. The counterweight mechanism is designed such that it can be mounted on the 1st link of the serial manipulator and coupled with link 2 using nylon thread. The weight is placed as per requirement of torque balancing. The free body diagram of mechanism can be seen in fig 6.

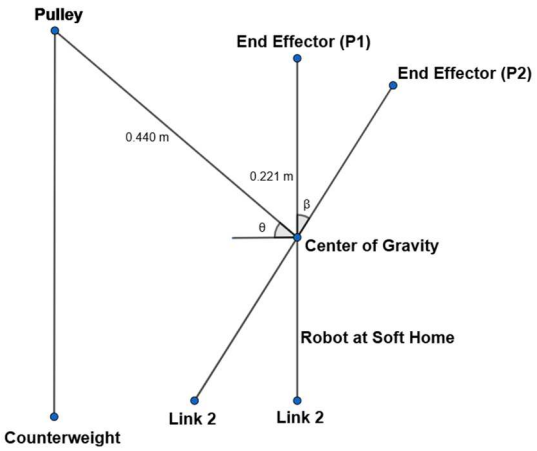

Fig 6: Free body diagram of counter weight mechanism

As the counterweight mechanism is attached to link 2, the length from link 2 to center of gravity of the rest of the structure is $0.221m$. The weight of the entire structure acting at the center of gravity is calculated to be $(2.41\ kg) \times (9.8\ m/s^2)$. Considering the dimensions of link 2, and the required torque at center of gravity, when the arm is oriented at angle β the torque can be calculated as:

$$T = 0.221 \times 2.41 \times 9.8 \times \cos(90 - \beta) = 5.219 \times \sin(\beta)$$

The holding torque of 2 motors of link 2 combined is $3.6N.m$. The counterweight applies torque in the opposite direction. The weight is 500 grams which means a force of $(0.5\ kg) \times (9.8m/s^2)$ will be applied. The angle of wire to the pulley will be θ.

$$T = 0.440 \times 0.5 \times 9.8 \times \cos(90 - \theta) = 2.156 \times \sin(\theta)$$

From experiments, the angle $\theta = 25$ degrees, when $\beta = 60$ degrees, the torque provided by counterweight is $2.025\ N.m$. As the motors torque and counterweight torque will be added hence, a total torque of $5.625\ N.m$ will be available. Final design and assembled mechanism can be seen in fig 7.

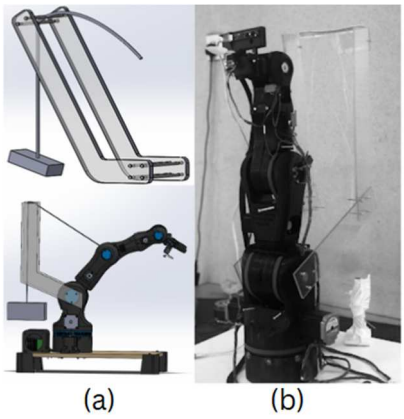

Fig 7: Counter weight mechanism a) CAD models b) Assembled with serial manipulator

B. Hybrid 3D Localization

To ensure the safe interaction between our assistive feeding system and the user's mouth accurately, 3D localization is paramount. We have integrated two complementary techniques: monocular depth estimation and sensor fusion with a laser depth sensor to accomplish this.

Monocular depth estimation is predicting the depth information of a scene from a single image or estimation of the distance of objects in a scene from a single camera viewpoint. We have used a pinhole camera model [14] based on the principle of triangulation to calculate monocular depth. The Pinhole camera as shown in fig 8 explains the mathematical relationship of the projection of points in 3D space onto an image plane [15].

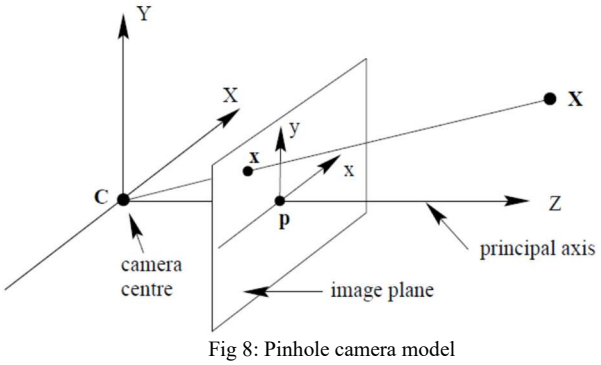

Fig 8: Pinhole camera model

Pinhole camera model works on the principle of similar triangles. In triangulation two image points and their matching camera projection rays are known, the 3D points that project onto those image points lies at the junction of the two rays

$$d = f_x \left(\frac{W}{w+1}\right)$$

$$Z - coordinate = d + f$$

As the target object is moved towards the screen the focal length is reduced and the "w" value is increased on the image plane and vice versa. In this case, we must know the actual height of an object whose depth estimation is required. For test we picked one person hence the distance between the eyes is the fixed parameter in our case.

We need focal length to proceed as per the equation, to obtain the focal length of the camera we performed camera calibration using checkerboard images. As a result of camera calibration intrinsic and extrinsic matrices are obtained.

So, dissecting the intrinsic parameters we get the following matrix [16]. The K matrix converts the camera coordinate system to a pixel coordinate system in two steps. 1) Camera to image 2) Image to pixel.

$$K = \begin{bmatrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{bmatrix}$$

Where fx and fy are the focal lengths of the camera in the x and y directions, respectively. The camera's optical centers, cx, and cy, define the primary point where the optical axis of the camera and the image plane cross. They depict any shifts or distortions caused by the lens and take into consideration the offset of the image center from the top-left corner [17].

$$k = \begin{bmatrix} 995.33 & 0 & 396.34 \\ 0 & 1002.32 & 265.16 \\ 0 & 0 & 1 \end{bmatrix}$$

The camera matrix K was obtained after performing calibration on checkerboard images from different poses and obtaining image points and object points.

As we already know the actual height of the object and the object whose depth is to be measured, then this 2D image point can be converted to the X and Y of the 3D world coordinates.

We assume that the pinhole camera model and the ground plane are parallel to the image plane. The formula to obtain X and Y of the 3D world coordinates from 2D image coordinates if the height of the object and object is known [18].

$$X = \frac{(x - c_x) \times Z}{f_x}$$

$$Y = \frac{\left((y - c_y) \times Z\right)}{f_y}$$

The camera is attached to the end effector of the robot and moves with the end effector. There becomes a situation where the camera is very close to the human face and the human face starts to fall out of the field of view of the camera. The face is not in the field of view of the camera and the detection of human face and tracking is lost. As there is no data for the robot in this situation to make the decision there is a possibility that the robot might hit the patient [19].

Table 1: Selection matrix for localization algorithm

No.	Algorithms	Limitations	Decision Points
1	Haar Cascade	1) Non frontal faces 2) Occluded faces	Rejected: a person can be in any position sitting or laying
2	Dlib HoG Face Detection	Increasing the resolution increases the delay	Rejected: higher resolution is required for better detection and further processing
3	OpenCV DNN	1) Non frontal faces 2) Massive occlusions	Rejected: Tracking will be affected due to missing detection of occluded face
4	Dlib DNN	Speed in real time due to large model size	Rejected: Slow speed and delays cause performance issues in real-time.
5	MediaPipe	Dependency Management	Selected: 1) Fast processing of video streams on computationally less expensive hardware 2) Prebuilt solutions that aid the processing further

To cater the aforementioned issue, we used a level 1 laser depth sensor which is biomedically safe to use in human-computer interactions. The depth will be shifted to the laser distance sensor when the face is not in the field of view of the camera.

For implementation of computer vision, we used Media pipe library based on selection matrix shown in table 1.

C. *Hybrid Kinematics Solution*

To solve the kinematic problem, we used Modified Denavit-Hartenberg (DH) Parameter. To reduce the complexity, we have excluded the 1 DOF of end-effector. Fig 9 shows the frame attachment of the robot as per modified DH parameters rules [17].

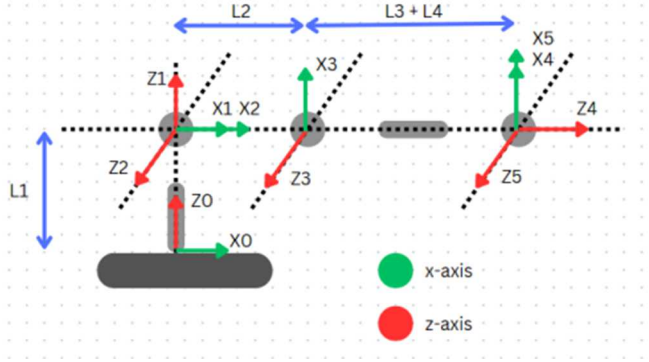

Fig 9: Frame attachment

The DH parameter values are given in table 2.

Table 2: DH Parameters

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	$L1$	$Q1$
2	90	0	0	$Q2$
3	0	$L2$	0	$Q3$
4	90	0	$L3+L4+L5$	$Q4$
5	-90	0	0	$Q5$

The forward kinematics equations were calculated by equating T_5^0 transformation matrix with unknown target matrix. Equation 1 was used to formulate T_5^0, while equation 2 is used to formulate the forward kinematics equations.

$$T_C^A = \mathrm{T}_B^A \times T_C^B \ (equ\ 1)$$

$$T_5^0 = \begin{bmatrix} r1 & r2 & r3 & p1 \\ r4 & r5 & r6 & p2 \\ r7 & r8 & r9 & p3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

To compute the joint angles or joint locations necessary to obtain a particular end-effector pose (position and orientation), robots use inverse kinematics. It is identifying the joint variables that adhere to the mechanical limitations placed on the robot as well as the geometric constraints.

We used algebraic method to find the closed form solution of joint angles. Equations of $p1$ and $p2$ are used to solve for $Q1$. $Q4$ was derived using equation of $r9$ and $Q5$ from $r7$ and $r8$. Final equations for $Q1$, $Q4$ and $Q5$ are as follows:

$$Q1 = \arctan2\left(\frac{p2}{p1}\right)$$

$$Q4 = \arcsin\left(\frac{r9}{-sin(Q2+Q3)}\right)$$

$$Q5 = \arcsin\left(\frac{r7 \times cos(Q2+Q3) - r8 \times sin(Q2+Q3) \times cos(Q4)}{(cos(Q2+Q3))^2 + (sin(Q2+Q3) + cos(Q4))^2}\right)$$

Algebraic solutions failed for Q2 and Q3. Paul's method was used as alternative using MATLAB which failed as well due to final equation being function of logarithmic integrals with no solution for aforementioned joint angles.

Based on results from algebraic and Paul's method, closed form solutions for these joint angles were not possible hence we used hybrid method and solved the remaining angles using numeral solution.

Simultaneous solution method was used to solve the remaining angles using equation of $p2$ and $p3$, and implemented on both MATLAB and python. Final implementation as on raspberry pi was on python using scipy library the results will be discussed in the results section.

$$p2 = \sin(Q1) \times (L3 \times \sin(Q2+Q3) + L2 \times \cos(Q2))$$
$$p3 = L1 - L3 \times \cos(Q2+Q3) + L2 \times \sin(Q2)$$

To get the inverse kinematics solution, target pose matrix is required which is given as T_W^B given as;

$$T_W^B = {T_W^C}^{-1} \times T_B^C$$

Where, T_W^C is world-camera transformation matrix and T_B^C base-camera transformation matrix.

In this project as previously mentioned, a 2D image is taken from the camera and computer vision algorithm detects the X_W and Y_W coordinates of the desired location then using a fix value of eye distance and pinhole camera model Z_W is estimated. These three are linear parameters of the target from the camera and the rotation parameters like pitch, yaw and roll are discarded because while feeding the spoon needs to be upright and the person needs to be in the same pose but there might be linear transformation which the robot will cover to feed. Hence keeping the assumption, the world to camera matrix can be given as:

$$T_W^C = \begin{bmatrix} 1 & 0 & 0 & X_W \\ 0 & 1 & 0 & Y_W \\ 0 & 0 & 1 & Z_W \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

IV. Results

In this section, we will discuss the results and limitations of our experiments.

A. Counter Weight Mechanism

Utilizing this mechanism reduced the torque requirements of link 2 motors, as given in table 3.

Table 3: Torque calculation at different angles

Sr. No	θ	β	T_C	T_R	T_M	T_{NET} $(T_C + T_M)$
1	15	30	0.55	2.6	3.6	4.15
2	30	60	1.07	4.5	3.6	4.67
3	60	90	1.86	5.2	3.6	5.46

θ is the angle between thread and horizontal plane
β is the angle of link 2
T_C is the torque provided by counter weight mechanism
T_R is the total torque required at that angle
T_M is the holding torque of joint 2 motors
T_{NET} is the net torque by using counter weight mechanism

Based on result we can conclude that this design provide dynamic counter torque based on the final angles at a particular orientation. Furthermore, the output is useful until a certain limit where counter weight is negative or less than the required value. Limitation of using this mechanism includes;

- Reduced number of solution for inverse kinematics due to mechanical limitations
- Work envelop is reduced due to mechanical limitations

B. Inservse Kinematics

The closed form solutions for joints Q1, Q4 and Q5 were found using the Algebraic Method. For the joints Q2, Q3, Paul's method and algebraic methods both failed due to involvement of logarithmic integrals. The simultaneous solution method was used to solve the remaining angles and implemented on both MATLAB and python. Final implementation as on raspberry pi was on python using scipy library.

Inverse kinematics has the following limitations:

- Simultaneous solution using numerical method are computationally more expensive than algebraic solutions

For validation of kinematics solution independently we took joint angles as $Q1 = 10^o$, $Q2 = 20^o$, $Q3 = 30^o$, $Q4 = 40^o$ and $Q5 = 50^o$, using forward kinematics the transformation matrix came out to be;

$$T_5^0 = \begin{bmatrix} -0.194 & -0.942 & -0.274 & 0.373 \\ -0.454 & 0.334 & -0.826 & 0.066 \\ 0.869 & -0.036 & -0.492 & 0.086 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

For T_5^0, the inverse kinematics solution came out to be:

$$[10 \quad 20 \quad 30 \quad 40 \quad 50]$$

C. 3D Localization

The fig 10 shows the accurate face detection, localization, and depth estimation implemented on the Raspberry Pi controller at 6 frames per second for smooth feeding. Table 4 shows the average error of $2.48cm$ with a maximum error of $3.75\ cm$ and a minimum error of $1.87cm$ between the actual depth and the computed depth using the camera.

Fig 10: 3D localization results on raspberry pi 3B

Table 4: Error Calculations

Actual Depth	**Computed Depth**	**Error (Computed – Actual)**	**%Error (Error × 100)**
46.00	48.17	2.17	4.71
51.00	52.87	1.87	3.60
73.00	75.62	2.62	3.58
40.50	42.50	2.00	4.94
60.0	63.75	3.75	6.25
	Average	2.48 cm	4.63 %

D. Assistive Feeding

With integration of counterweight mechanism, inverse kinematics based control and hybrid localization technique we were successful in feeding rice. The system was tested by

feeding 3 different person. The results of feeding can be seen in fig 11.

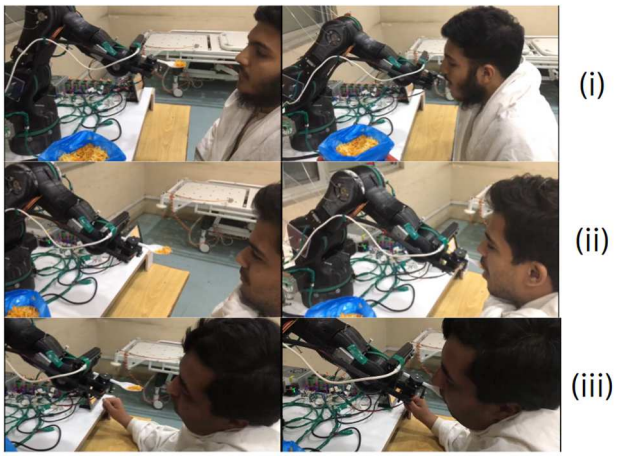

Fig 11: Assistive feeding system feeding rice to 3 different test subjects

Based on post-feeding feedback from care receivers, although the system successfully achieved its main objective of feeding food, there remain areas for improvement, particularly in the speed of the feeding cycle. This limitation was primarily due to the processing power of the embedded system (i.e., Raspberry Pi 3B) used in our device.

V. Conclusion

Assistive Feeding System is quite complex system involving not only the requirements like precision, accuracy and repeatability but due to human interaction it has to be collaborative and should have maximum safety features and fail safes. In case of active feeding approach the system needs to be even more robust and incorporate anomaly detections. We were successfully able to implement active feeding using hybrid methodologies on custom low cost hardware, but there are still areas to work on including but not limited to anomaly detection (i.e. choking, and physical hitting), utilization of control mode without physical hardware interaction (i.e. voice control, and facial expression based control), and dynamic scoping (i.e. different bite sizes etc.).

Acknowledgment

We thank BCN3D for open source design of robotic manipulator. This work was supported by National Grassroot Research Grant 2023, National Center of Robotics and Automation, and Department of Mechatronics Engineering College of Electrical and Mechanical Engineering, NUST.

References

[1] "World Report on Disability," Jun. 20, 2024. Available: https://www.who.int/teams/noncommunicable-diseases/sensory-functions-disability-and-rehabilitation/world-report-on-disability/

[2] A. to A. T. and M. Devices, "Global report on assistive technology: summary," Dec. 08, 2022. Available: https://www.who.int/publications/i/item/9789240049178

[3] International Council of Nurses, 2021, The Global Nursing Shortage and Nurse Retention, (https://www.icn.ch/sites/default/files/inline-files/ICN%20Policy%20Brief_Nurse%20Shortage%20and%20Retention.pdf) (Accessed 13 November 2023)

[4] P. Rashidi and A. Mihailidis, "A survey on ambient-assisted living tools for older adults," *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 3, pp. 579–590, 2012.

[5] A. Chiò, A. Gauthier, A. Vignola, A. Calvo, P. Ghiglione, E. Cavallo, *et al.*, "Caregiver time use in ALS," *Neurology*, vol. 67, no. 5, pp. 902–904, 2006.

[6] P. Lopes, R. Lavoie, R. Faldu, N. Aquino, J. Barron, M. Kante, *et al.*, "Eye-controlled robotic feeding arm technology," presented at [conference/journal details missing, ensure to add conference or publisher information], 2012.

[7] Secom, Meal-assistance robot, my spoon, 2017, https://www.secom.co.jp/english/myspoon/. (Accessed 18 September 2022)

[8] Mealtime Partners, Specializing in assistive dinning and drinking equipment, 2017, http://www.mealtimepartners.com/. (Accessed 18 September 2022)

[9] Patterson Medical, Meal Buddy, 2017 http://pattersonmedical.com/. (Accessed 18 Septemper 2022)

[10] Daehyung Park, Yuuna Hoshi, Harshal P. Mahajan, Ho Keun Kim, Zackory Erickson, Wendy A. Rogers, Charles C. Kemp, Active robot-assisted feeding with a general-purpose mobile manipulator: Design, evaluation, and lessons learned, Robotics and Autonomous Systems, Volume 124, 2020, 103344, ISSN 0921-8890, https://doi.org/10.1016/j.robot.2019.103344. (https://www.sciencedirect.com/science/article/pii/S0921889018307061)

[11] I. Naotunna, C. J. Perera, C. Sandaruwan, R. A. R. C. Gopura and T. D. Lalitharatne, "Meal assistance robots: A review on current status, challenges and future directions," 2015 IEEE/SICE International Symposium on System Integration (SII), Nagoya, Japan, 2015, pp. 211-216, doi: 10.1109/SII.2015.7404980. keywords: {Robot sensing systems;Mobile robots;DC motors;Keyboards;Sociology;Statistics}

[12] Jorge. (2022, September 12). *BCN3D MOVEO: A fully Open Source 3D printed robot arm*. BCN3D Technologies. https://www.bcn3d.com/bcn3d-moveo-the-future-of-learning-robotic-arm/

[13] Chi, M., Liu, Y., Yao, Y. *et al.* Development and evaluation of demonstration information recording approach for wheelchair mounted robotic arm. *Complex Intell. Syst.* **8**, 2843–2857 (2022). https://doi.org/10.1007/s40747-021-00350-9

[14] PinHole Camera model , https://www.dsi.unive.it/~bergamasco/teachingfiles/cvslides/11_pinhole_camera_model.pdf

[15] D. Tan, "Inverse Projection Transformation - towards Data science," *Medium*, Dec. 13, 2021. Available: https://towardsdatascience.com/inverse-projection-transformation-c866ccedef1c

[16] "Image formation," *Robot Academy*. Available: https://robotacademy.net.au/lesson/image-formation/

[17] Stanford University [Online] Available at: https://web.stanford.edu/class/ee364a/lectures/functions.pdf

[18] Edmund Optics, "Understanding focal length and field of view," *Edmund Optics*. Available: https://www.edmundoptics.com/knowledge-center/application-notes/imaging/understanding-focal-length-and-field-of-view/

[19] Princeton Instruments, "Learn | Field of View & Angular Field of View | Teledyne Princeton Instruments," *Teledyne Princeton Instruments*, Oct. 16, 2020. Available: https://www.princetoninstruments.com/learn/camera-fundamentals/field-of-view-and-angular-field-of-view