drones

MDPI

Article

Deep Learning-Based Cost-Effective and Responsive Robot for Autism Treatment

Aditya Singh [1,*], Kislay Raj [2], Teerath Kumar [2], Swapnil Verma [3] and Arunabha M. Roy [4,*]

[1] Center of Intelligent Robotics, Indian Institute of Information Technology, Allahabad 211015, India
[2] SFI for Research Training in Artificial Intelligence, Dublin City University, D09 Dublin, Ireland
[3] United Kingdom Atomic Energy Authority, Abingdon OX14 3DB, UK
[4] Aerospace Engineering Department, University of Michigan, Ann Arbor, MI 48109, USA
* Correspondence: rsi2018003@iiita.ac.in (A.S.); arunabhr.umich@gmail.com (A.M.R.)

Abstract: Recent studies state that, for a person with autism spectrum disorder, learning and improvement is often seen in environments where technological tools are involved. A robot is an excellent tool to be used in therapy and teaching. It can transform teaching methods, not just in the classrooms but also in the in-house clinical practices. With the rapid advancement in deep learning techniques, robots became more capable of handling human behaviour. In this paper, we present a cost-efficient, socially designed robot called 'Tinku', developed to assist in teaching special needs children. 'Tinku' is low cost but is full of features and has the ability to produce human-like expressions. Its design is inspired by the widely accepted animated character 'WALL-E'. Its capabilities include offline speech processing and computer vision—we used light object detection models, such as Yolo v3-tiny and single shot detector (SSD)—for obstacle avoidance, non-verbal communication, expressing emotions in an anthropomorphic way, etc. It uses an onboard deep learning technique to localize the objects in the scene and uses the information for semantic perception. We have developed several lessons for training using these features. A sample lesson about brushing is discussed to show the robot's capabilities. Tinku is cute, and loaded with lots of features, and the management of all the processes is mind-blowing. It is developed in the supervision of clinical experts and its condition for application is taken care of. A small survey on the appearance is also discussed. More importantly, it is tested on small children for the acceptance of the technology and compatibility in terms of voice interaction. It helps autistic kids using state-of-the-art deep learning models. Autism Spectral disorders are being increasingly identified today's world. The studies show that children are prone to interact with technology more comfortably than a with human instructor. To fulfil this demand, we presented a cost-effective solution in the form of a robot with some common lessons for the training of an autism-affected child.

Keywords: autism therapy; computer vision; robot–child interaction; robot design; robot vision

check for updates

Citation: Singh, A.; Raj, K.; Kumar, T.; Verma, S.; Roy, A.M. Deep Learning-Based Cost-Effective and Responsive Robot for Autism Treatment. *Drones* **2023**, *7*, 81. https://doi.org/10.3390/drones7020081

Academic Editor: Diego González-Aguilera

Received: 17 November 2022
Revised: 7 January 2023
Accepted: 15 January 2023
Published: 23 January 2023

1. Introduction

Deep learning (DL) algorithms have shown promising results in the range of domains, such as images [1–10], audio [11–16], text [17,18], object detection [19–21], brain–computer interface [22–24], and across diverse scientific disciplines [25,26]. Robotics can utilize a combination of these domains and it can transform the way of living life. It has already been impacting many areas of our lives, and one of that areas is education. By the addition of deep learning models to robot functions, robots can make learning more practical, exciting, and fun [27]. It is one of the advanced and interdisciplinary tools in STEM education [28]. There are many commercial robots available in the market to teach programming, mathematics, sensor technology, mechanics, and electronics, among many other things [29]. The ease of learning has also increased with the contribution of open-source, low-cost, easy-to-use tools, such as single-board computers and micro-controllers. Learning through robots also increases collaboration and group management. Robotics has made an immense contribution to the field of education, but it is mainly limited to universities and research institutes. Very few schools across the globe have an access to any highly effective robot. One main reason for the absence of a robot in the school is the high cost [30]. It can also be used for clinical practices at a mass level to train special needs persons similarly to with the autism-affected ones. People with disabilities are a group whose quality of life could be improved with the assitance of technology Social robotics is one of the ways through which they can have assistance in learning, doing day-to-day tasks, playing, and many more things. This project aims to design a social robot that will interact and assist in teaching day-to-day tasks to children with an intellectual impairment. With the inclusion of deep learning models, it can handle very sophisticated high-level applications such as recognition and detection. This platform can also be used for children that have an average intellectual level. The main challenges of this project are:

1. Design a cost-effective social robot that is robust and friendly.
2. Design a lesson for teaching day-to-day tasks to the intellectually impaired student.
3. Implement efficient deep learning models on the development boards for better perception of the robot.
4. Utilize the power of a single board controller for the use of deep learning models.

In this paper, deep neural networks (DNNs) are well suited for use with robots because they are flexible and can be used in structures that other machine learning models cannot support; applying deep learning to robotics is an active research area. Humans must be willing to give up some control if machine learning is used to control robots. At first, this may seem counter-intuitive, but doing so allows the system to start learning independently. Due to its adaptability, the system has the potential to make better use of the guidance provided by humans eventually. We present a low-cost social robot called Tinku. Tinku is designed focus on education and children. Its design is inspired by the animated character WALL-E whose popularity inherently gives it a nice look. Inside, the Tinku has many features, due to its single board computer powerful enough to handle computer vision [1–3,31] and speech recognition-like tasks [11,13,32]; its low power but sensor-friendly Arduino microcontroller; and lots of sensors, such as ultrasonic sensor, camera, mic, and motor encoder. It also has motors for locomotion, a touchscreen that acts as its faces and is also used for output devices while teaching, and a neck like a servo assembly on which the touch screen is mounted. The sole purpose of the neck assembly is to complement the emotional expression and non-verbal communication exhibited by Tinku. It can be used to teach robotics, mathematics, electronics, programming, engineering, etc. One of the most critical and novel features of Tinku is its ability to teach STEM-related subjects and everyday activities, such as brushing, storytelling, and table manners, among other things, thanks to its face-like touchscreen and ability to express emotional expressions. Tinku is not just another educational robot, it is a highly cost-effective social robot developed for use in universities and primary schools. The demonstration videos for different activities are available at https://www.youtube.com/watch?v=wkXJ1MAwCWc&list=PL2-eMlAlKTtUkLUnikJZctZzdpKxxX0R4&index=2 (accessed on 16 November 2022). All the instructions for development are available on the repository at https://www.hackster.io/usavswapnil/offline-speech-processing-82c506 (accessed on 16 November 2022).

The rest of the work is organized as follows: Section 2 discusses closely related work; Section 3 describes the required hardware, which includes the designs and electronics of the prototypes; Section 4 describes the software architecture and information flow during the processing of activities; Section 6 includes the results of its application for various activities and discusses the performance; and, finally, Section 7 concludes the whole paper.

2. Related Works

Social robotics is an emerging area of research in the field of robotics. In the last few years, this field also proliferated due to the rapid evolution in artificial intelligence and machine learning. Much research [33] has been performed on using social robots in education [29,34]. Similar research has been completed by A. Billard et al. [35], where the authors talked about how a mini-humanoid doll-shaped robot is helping to teach normal and cognitively impaired children through educational and entertaining games. Daniel et al. [36] presented a study showing that the interaction of autism-affected children is more favourable towards a robot than a human teacher. Author research indicated that toy robots can be a better tool for teaching children. The author uses games such as *Robota learns to dress* to teach children who have autism to dress themselves appropriately. Another game called *Robota drawing game* provides an incentive for the child to draw straight lines and measure the child's ability to do so. C. Breazeal [37] worked on the non-verbal communication between a human and a robot, where the author researched the efficiency of the task performed, robustness to errors, and transparency and understandability of the robot's internal state. Then the author proved it using an experimental setup where a human subject guides the robot to perform a physical task using speech and gesture. Terrence Fong et al. [38] surveyed socially interactive robots. The authors state that there are two types of social robots. (a) The biologically inspired robot [39], which internally simulates or mimic the social behaviour or intelligence found in living creatures. (b) The functionally designed robot outwardly appears to be socially intelligent, even though the internal design does not have any basis in nature. Cory D. Kidd [40] worked on the effect of a robot on user perception, which concludes that the robot can be a partner rather than a tool in the sectors such as education and entertainment because of their potential to be perceived as trusting, helpful, engaging, and reliable. Furthermore, author also concluded that the robot's physical appearance and presence greatly influence the user's perception of these characteristics. In her paper towards sociable robots, C. Breazeal [41] classifies social robots into four categories: socially evocative, socially communicative, socially responsive, and sociable. The main difference between these categories is their ability to support the human social model in complex environments. Breazeal demonstrated these differences by presenting a case study of Kismet, focusing on vocal turn-taking behaviour. Fatemeh Maleki [42], in his research, studied robots in animated movies and created a rule set that describes friendly, socially acceptable, kind, and cute robots. Based on all these studies and research, the design of Tinku is finalized.

There are numerous examples of using robots in the human environment for medical therapy, education, and entertainment. One example is a school in Birmingham, Topcliffe Primary [43]. This initiative is carried out by Dr Karen Guldberg, director of the University of Birmingham's Autism Centre for Education and Research and Aldebaran Robotics.

3. Hardware

The initially proposed methodology used an existing robotic platform and designed a software architecture consisting of lessons for teaching day-to-day tasks. The first robot platform for this project was NAO, which is a humanoid robot developed by the French company Aldebaran Robotics. Many schools and universities are already using it for research and education. One such example is a school in Birmingham, Topcliffe Primary. The design of the NAO, its diverse sets of sensors and the software package provided by Aldebaran makes it one of the best contender for this project. However, all this luxury comes at a cost. About this project, the actual cost of the robot and the scalability. To overcome these limitations, we decided to design the robot from scratch, which has all the features required for this project and is relatively cost-effective at the same time.

3.1. Design

The design of the robot is inspired by an animated character *WALL-E* featured in the movie by the same name. Inspiration is taken from WALL-E due to its anthropomorphic design and wide acceptance among children and adults. The robot for this project is meticulously designed to fulfil the current requirements, keeping future modifications and improvements in mind. We call it ***'Tinku'***. Tinku has a screen in place of the eyes of the

WALL-E. The screen is its primary output device for all kinds of visual communication. The body of the Tinku is also smaller than that of WALL-E. It is designed to be in proportion to its head so that it does not look weird or scary. For locomotion, it has two DC motors assembled in differential drive mode. The tracks and wheels are also smaller to complement the design of the robot. Tinku has two versions; in the first version, the material used for the robot's body is acrylic plastic. The reason for using this material is because it is transparent and through which the robot's internal electronics can be seen, which helps to teach electronics. The main disadvantage of using acrylic plastic is that it can not handle tension or compression and is prone to cracking. The first version was also more prominent, taller, and heavier, resulting in inertial slippage concerning the ground while moving. To eliminate these disadvantages, Tinku was redesigned. In the second version, the material used for the body was wood and steel, but some parts are still made of acrylic plastic, through which the internal electronics can be seen. The overall volume of the body was reduced, and the distance between the wheels was also reduced to induce more tension in the tracks, resulting in better locomotion. Tinku does not have hands or any other manipulator to manipulate its environment but can be easily included in future modifications. The new version of the Tinku is smaller in size, less prone to steering casualty, and better looking than the older version.

One of the important features of Tinku, which also makes it more anthropomorphic, is its capacity for non-verbal communication and emotional expression through moving its head. Its neck powers these specific head movements similarly to as in humans. Its neck has three servos which provide three degrees of freedom. Three degrees of freedom are enough for most types of head gestures. Tinku can say *yes* and *no* through head gestures and can also express different types of emotions, such as *sadness*, *confusion*, etc. The base servo enables a *yaw* motion which is essential for saying *no*, the middle servo enables the *roll* motion which is essential for the *confusion* expression, and the end servo enables the *pitch* motion, which is essential for saying *yes* or showing similar head gestures.

3.2. Electronics

The sensors fused in the Tinku are ultrasonic sensors, a camera, mic, and encoders. Ultrasonic sensors are used to avoid obstacles while going from point A to point B. A total of six ultrasonic sensors are placed around the body of the robot. The camera is used for tracking the faces and collecting the visual information about the surrounding. The mic is used in speech processing. The encoders are used to measure the distance traveled by the robot. For connecting the ultrasonic sensor and motor encoders with the micro-controller, a custom interfacing PCB is used. It also has an 11.1v lithium polymer battery, with a capacity of 3000 mAh. The overall specification is discuseed in Table 1.

The actuators are DC motors for locomotion and servo motors for the head gestures. DC motor driver is based on the L298n, a dual full-bridge motor driver IC. As the name suggests, it can control two DC motors individually. For controlling the servo motors, a custom servo driver circuit is used. Figure 1 shows the schematics of the dynamixel servo driver. The IC used in the servo driver is the 74HC244, an octal buffer and line driver IC with the 3-state outputs. This PCB is a way to control the servos, which act as the neck. It has a 1 GHz quad-core CPU, 1 GB of RAM and micro-SD card support. In terms of connectivity, it has Wi-Fi and Ethernet. Tinku also has a 7-inch touchscreen, its primary output device for visual communication. This screen acts as its face.

Table 1. Specifications of Tinku.

Processor	NXP® i.MX 6 ARM Cortex-A9 CPU quad core 1 GHz
RAM	DDR3, 1 GB
ROM	8 GB micro SD card
Connectivity	Ethernet RJ45 and Wifi module
OS	UDOObuntu 2.0
Microcontroller	ATmega 1280
Camera and Mic	Logitech c-270
Screen	Waveshare 7″, 800 × 480 touch screen
DC Motor	Side shaft, high torque motor
Motor Driver	L298n IC
Servo Motor	Dynamixel AX-12+
Sensors	Ultrasonic sensor, Motor encoder
Power	11.1v, Lipo battery, 3000 mAh

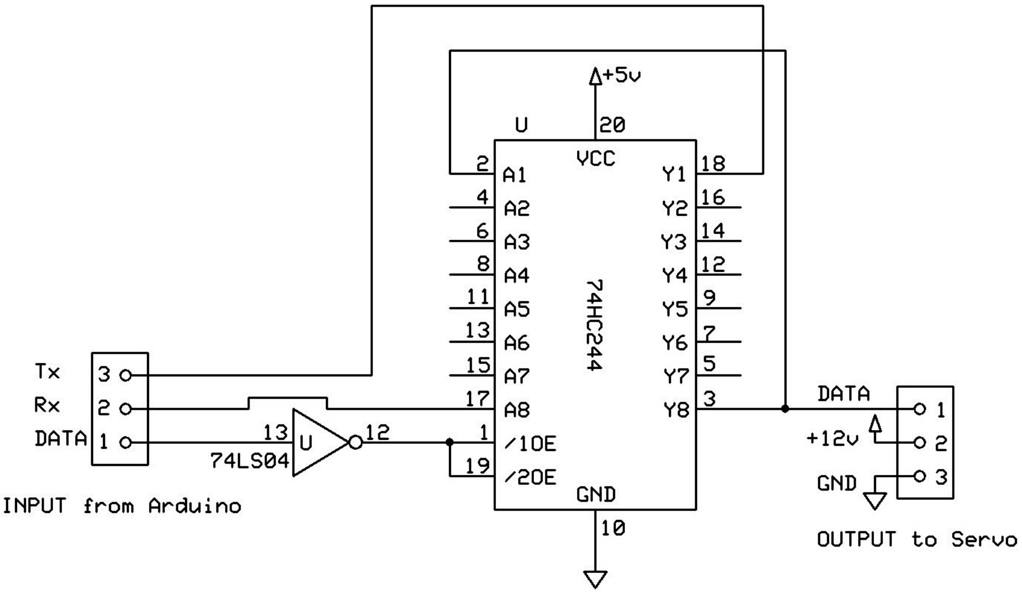


Figure 1. Schematics of dynamixel servo driver.

4. Software

The single-board computer has its own full-fledged Linux-based OS called UDOObuntu 2.0 and the Arduino uses single loop-based code to achieve all its tasks. Two different languages are used to program the robot, Python and C/C++. Python is used for high-level tasks such as speech and vision processing, visual communication, etc. C/C++ is used in Arduino to handle tasks related to sensors and actuators.

4.1. Speech Processing

One of the interaction methods implemented in the Tinku is offline speech processing. It is not a complete speech processing application, like with an Alexa or Google Assistant, but it is a hot word detection method, similar to an *OK Google*. This hot word detection is used in such a way that it detects a complete sentence. It is not a robust mechanism and has lots of space for improvement in future development.

4.2. Computer Vision

Another non-tangible interaction method implemented in Tinku is vision processing. It uses an open-source computer vision library, OpenCV, for this purpose. It detects the face, mouth, and colour. It also captures an image for the given command. The implementation of vision-related features fully fills the requirements of the sample lessons.

For more specific detection, a TF-Lite-based [44] object detection model is used. It uses an SSD network [45] model for object detection. We have compared Tiny-YOLO and TF-lite models for the robot's performance on our development board. TF-lite model is trained on the COCO dataset, which has 80 object classes. So in an extended application, we can use additional class labels. It will have more features in future development.

Initially, regional convolutional neural network (RCNN) [46] was proposed for object detection. The main drawback RCNN was it extracts 2000 regions are extracted per image leading to speed issue—40 to 50 seconds per image. Then Fast-RCNN [47] was proposed to deal RCNN speed issue, it replaces three CNN with single CNN. The Fast-RCNN speed is better than RCNN but in real-time datasets, it still lacks speed as Fast-RCNN uses selective search, which takes a lot of time. Another version of RCNN, Faster-RCNN [48] was proposed to deal Fast-RCNN speed issue. The key difference between Fast-RCNN and Faster-RCNN is that former uses selective search and later uses region proposal network. All the discussed networks are two-stage detectors. Although these are accurate they are very slow. As a trade-off between accuracy and speed, different single models, such as Yolo versions, SSD, and many more are proposed [49]. The used models in the paper—Yolo V2-tiny and single shot detector are discussed below:

4.2.1. Single-Shot Detector (SSD)

Single-shot detector (SSD) [4,50,51] have an excellent trade-off between accuracy and speed. SSD is applied single time to detect the object(s) in the image. It exploits the anchor boxes, such as Faster-RCNN, at different aspect ratio and learns offset in place of determining the boxes. First single image is fed to SDD, which uses VGG16 architecture as backbone. At the end of VGG, extra features layers are added, which are downsampled in a sequential manner. The overall architecture of SSD is shown in Figure 2.

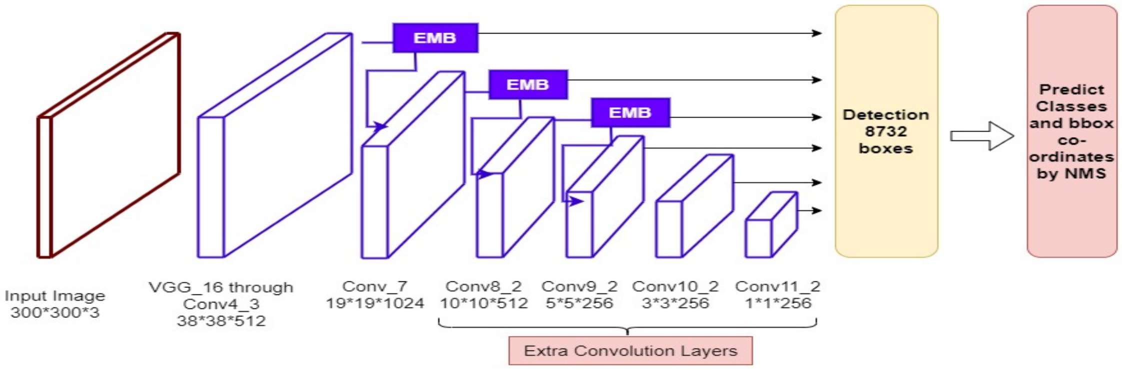


Figure 2. Single shot detector architecture [50].

4.2.2. Yolo V3-Tiny

Yolo divides image into grid of random size [52]. It predicts the offset of the default box for each location of feature map unlike SSD. Yolo V3-tiny is another lighter version of Yolo, is the decreased depth of the convolutional layer. It was introduced by Joseph Redmon [53]. The reason for choosing the model, it is a lighter version and much faster (almost 442% faster) than previous Yolo versions but with scarified detection accuracy. The Yolo v3-tiny model uses Darknet-53 architecture with many 1 × 1 convolution layers along with 3 × 3 convolution layers with aim of extracting features. The model predicts three dimensions tensors, first for objectness score, second for bounding box, and third for class predictions at two different scales. For the final prediction, we ignore the bounding boxes having score less than a certain threshold. For more detail, Yolo v3-tiny architecture is shown in Figure 3.

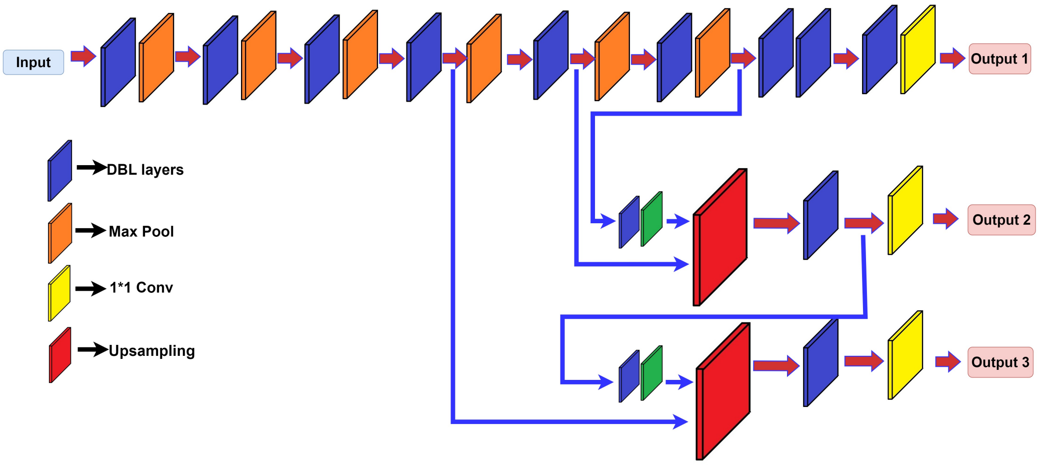


Figure 3. Yolo V3-tiny architecture [54].

4.3. Non-Verbal Communication

Facial expression and body language play a major role in communication between people. Many times we humans communicate non-verbally and only using our facial expressions and body language. Tinku has this capability of non-verbal communication. This skill makes it more anthropomorphic. Tinku uses its *neck* having three degrees of freedom to manipulate its *head* in order to communicate non-verbally. To complement its head action, it displays images and gifs on its screen. In the same way, it also expresses emotions. Tinku has the ability to express sadness, joy, and confusion.

4.4. Obstacle Avoidance

For a mobile robot, an important feature is obstacle avoidance. To avoid obstacles, Tinku has six ultrasonic sensors, four in the front, and two in the back. Only sensors have depth sensing capability. The ultrasonic sensor is good for distance calculation, but it is not good for mapping the environment. The lack of a mapping sensor introduces some limitations of using a complex path planning and obstacle avoidance algorithm. The obstacle avoidance algorithm used here is reactive and very basic in nature, but it does a decent job of avoiding the obstacles.

Figure 4 shows the flowchart of the architecture of the obstacle avoidance algorithm. Here the robot keeps driving until the goal is reached. If it finds any obstacle in its way, then it takes action according to the position of the obstacle in front of the robot. If the obstacle is on its left, the robot takes a right turn; if it is on its right or middle, then it takes a left turn. If the robot is surrounded by the obstacle, then it moves in the backward direction until it reaches free space. Once it avoids the obstacle, it again follows its goal.

4.5. Teaching

Teaching is difficult irrespective of the importance of a lesson or task; a child will not pay attention to it unless it is a fun to do. This task becomes more complicated than it already is when teaching an intellectually impaired child. To make the lesson engaging, it needs to be playful and fun for the child.

In this project, we prepared the lessons for *brushing*. The first thing to do while preparing a lesson for an intellectually impaired child is to break the task into multiple steps. The steps involved in the brushing are

1. Identify the toothbrush.
2. Pick up the toothbrush.
3. Hold it properly in the left/right hand.
4. Rinse the toothbrush.
5. Apply toothpaste on the toothbrush bristle.
6. Put the bristle in your mouth.
7. Move the brush gently in all directions to clean your teeth.
8. Brush for at least 3 minutes.
9. Do not intake the foam and spit it out.
10. Rinse your mouth properly.
11. Rinse your brush properly.
12. Put the brush in its place.

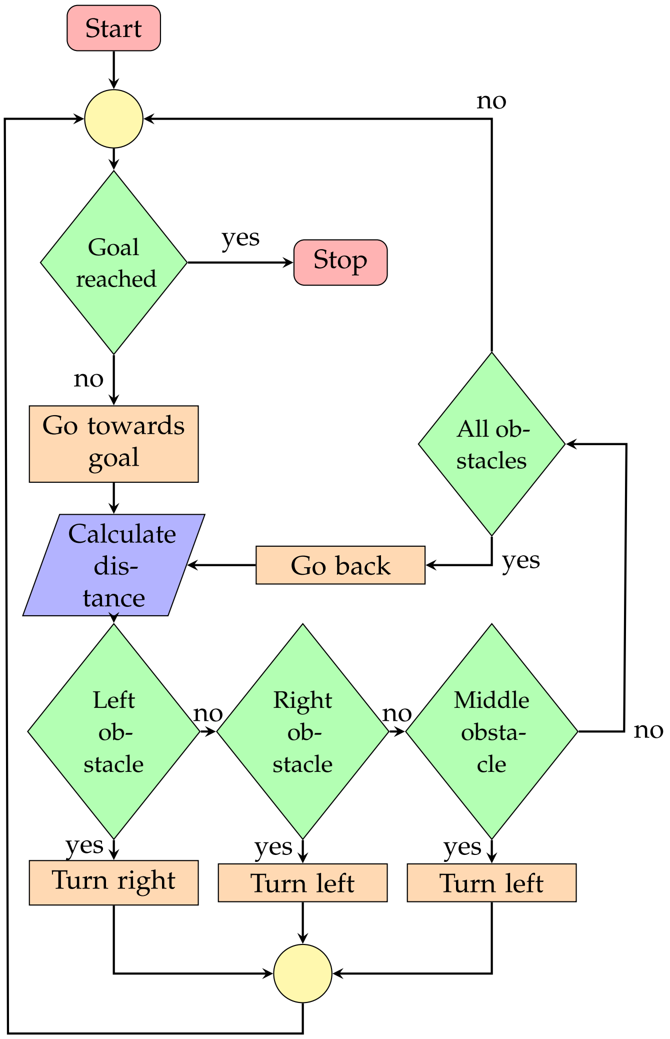


Figure 4. Flow chart of obstacle avoidance algorithm.

To simplify the teaching process, We divided the task into 3 lessons.

- Lesson 1: Identify the brush.
- Lesson 2: Hold the brush properly.
- Lesson 3: Brushing.

The first chapter in brushing is about identifying the brush. To teach a student what a brush looks like, Tinku will display different types of toothbrushes one by one on its screen. By seeing the images of the toothbrush and engaging with the robot, a student can learn what a toothbrush looks like and what it does. For Chapter Two, the same strategy is followed. Chapter Two is about how to hold the toothbrush correctly. In this chapter, Tinku shares the images consisting of steps depicting the correct hand gesture for holding the brush. Chapter Three is about how to do the actual brushing. In this chapter, Tinku instructs the student about how to put the brush in the mouth and perform the various hand motions for effective brushing.

Some form of feedback is necessary to understand how much the student has learned. This lesson also has feedback methods in the form of tests. In Chapter One, Tinku shows some random pictures and asks the student whether that image is of a brush or not. If the student gives a correct answer, Tinku expresses joy. If the student gives a wrong answer, then it expresses sadness. Chapter Two takes help from a human teacher in order to identify whether the student is holding the brush correctly or not. The feedback method of Chapter Three uses a computer vision algorithm. It automatically tracks the mouth of the student and the brush and continuously monitors whether the student is brushing or not. If the student is brushing, then it provides positive feedback and expresses joy, and if the student is not brushing, then it expresses sadness and encourages them to perform the brushing task. Figure 5 represents the architecture of the lesson. By default, the Tinku is in sleeping mode. To wake up, it needs to hear its name, *Tinku*. Then it waits for the voice inputs upon which it will react. If the voice input is *start lesson* then it will start the lesson and if the voice input is *exit* then it will exit the lesson. After starting the lesson, it waits for menu commands. Menu commands consist of *learn, test* and *exit* modules. Voice input for the *learn* module is *image*. In learn module, it displays the images or any other media based on the respective chapters. To navigate in this module, the commands are *next, back*, and *stop*. On detecting *next* it goes to the next media and on detecting *back* it displays the previous one. The module will not stop until it hears *stop*. On detecting *stop* it goes back to the menu commands where again the options are *learn, test*, and *exit*.

To enter the *test* module the command is *test*. In the test module, different feedback methods will be executed based on the chapter. This process repeats until it hears *stop*.

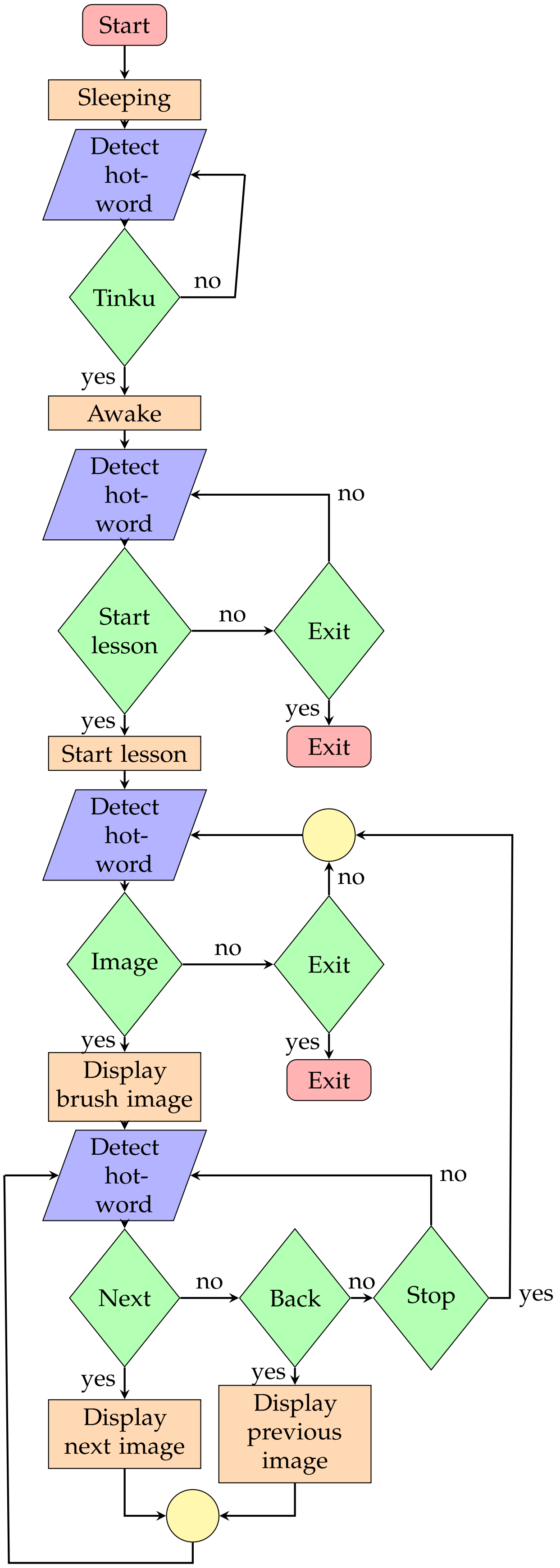


Figure 5. Flow chart of the lesson.

5. Configuration and Training

The proposed setup uses deep learning models for purposes, such as object detection and speech processing. The models used have been chosen while considering their latency on the development board is the TF-lite model for object detection. It is a pre-trained model on the COCO dataset, which includes 80 object classes. Our main focus was a human object from the pre-trained model. Using 0.5 thresholds for a person object, we have detected the person class only.

For speech processing, we have used speech-to-text Google API. This API is easy to use and supports to python platform, which makes it compatible. This service is also using pre-trained models and these are easy to use.

6. Results and Discussion

Before discussing the result, let us look at the challenges of this project.

1. Design a cost-effective and social robot which is robust as well as friendly looking.
2. Design a lesson for teaching day to day task to the intellectually impaired student.

The challenge number 1 is related to hardware problem and the challenge number 2 is related to a software problem.

6.1. Challenge 1: Hardware

Lets breakdown the challenge 1 into smaller parts.

- Design a social robot.
- It should be cost effective.
- It should be friendly looking.
- It should be robust.

6.1.1. Social and Friendly Design

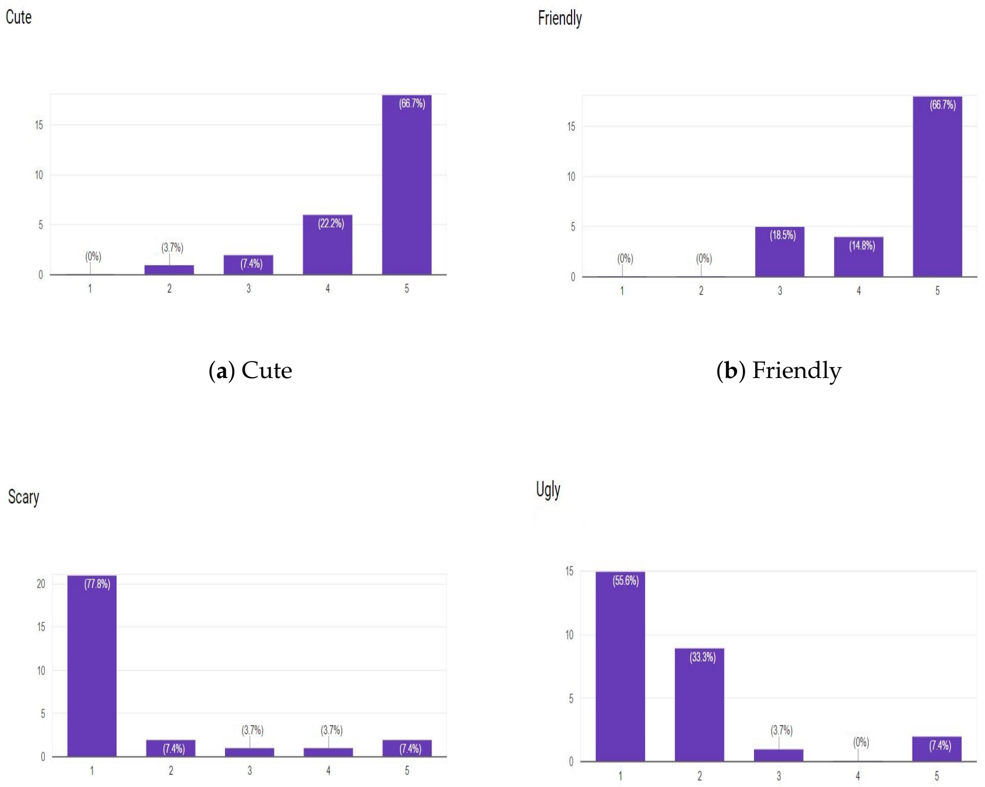
The very first problem is to build a robot from scratch. Making a robot from scratch has advantages relating to cost, design, functionality, etc. To design the robot as social and friendly looking, the inspiration is taken from toys, animated movie characters, and cartoons because children love these things and adults do too. Finally, the animated movie character WALL-E is used as the main design inspiration. WALL-E is used because it is also a robot in the movie; it is a widely famous character that adults and children love and has a practical design to replicate. Other animated movie characters such as *Baymax* from movie *Big Hero 6* and *Eve* from the movie *WALL-E* have a more social design, but they are impractical to build. Expressing emotion is essential for a social robot because it is a way of connecting with humans. Humans express emotions through their faces and body language. WALL-E has two cameras which are designed as its eyes. In the movie, he expresses most of his emotions using his eyes and head gestures. However, Tinku has a screen acting as his face and head. To express emotion, it needed a face, where Baymax comes in. The face of the Baymax is simple and has only eyes connected by a line. Baymax also expresses lots of his emotions in his eyes. So in the design of Tinku, inspiration was taken from both WALL-E and Baymax.

Figure 6 shows the result of the survey done on the appearance of the Tinku. The sample size was 60 and it belongs to both genders with a variety of age from 15 to 62 years. It is tested 16 on small children for the acceptance of the technology and compatibility in terms of voice interaction. It helps autistic kids using state-of-the-art deep learning models. Figure 6a represents the cuteness of the Tinku, where 66.7% of total candidates who participated in the survey rated it 5 stars, 22.2% of candidates gave 4 stars, 7.4% of candidates rated it 3 stars, 3.7% of candidates rated 2 stars and 0% of candidate rated 1 star. So clearly Tinku is cute.

Figure 6b represents how *friendly* Tinku look, where 66.7% of total candidate who participated in the survey rated it 5 stars, 14.8% of candidates rated it 4 stars, 18.5% of candidates rated it 3 stars and 0% of candidate rated it 2 stars and 1 star. So clearly Tinku looks friendly.

Figure 6c represents how *scary* Tinku looks, where 7.4% of total candidate who participated in the survey gave him 5 stars, 3.7% of candidates gave 4 stars, 3.7% of candidates gave him 3 stars, 7.4% of candidates gave 2 stars and 77.8% of candidate gave 1 star. So clearly Tinku does not look scary.

Figure 6d represents how much *ugly* Tinku look, where 7.4% of total candidate who participated in the survey rated it 5 stars, 0% of candidates gave 4 stars, 3.7% of candidates rated it 3 stars, 33.3% of candidates rated 2 stars and 55.6% of candidate rated 1 star. So clearly Tinku does not look ugly too.



(**a**) Cute (**b**) Friendly

(**c**) Scary (**d**) Ugly

Figure 6. A survey report on appearance of Tinku. Note: The sample size was 60 and it belongs to both genders with variation of age from 15 to 62 years.

6.1.2. Cost

To reduce the robot's cost, the selection of components and materials plays an important role. Some PCBs, such as sensor connectors and servo drivers, are designed and manufactured in-house to reduce the project's cost. In place of using a camera and mic as separate modules for vision and speech processing, a webcam is used, which has both camera and mic in one module; this also reduces the cost as one less component is being used.

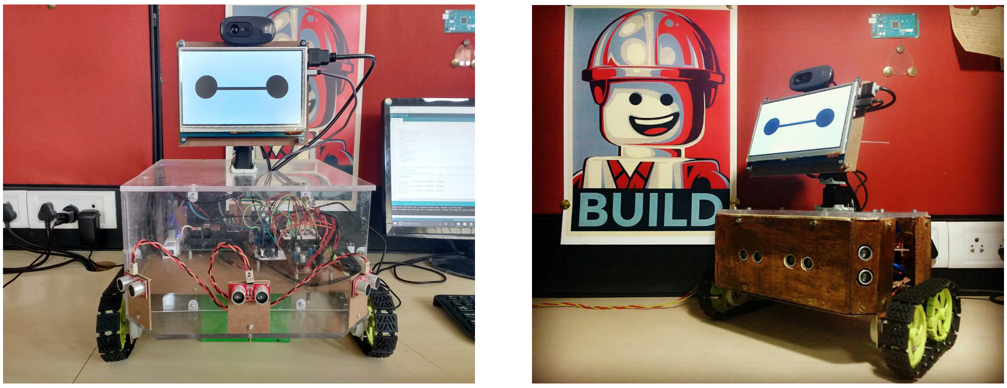
From the Table 2, the total cost of the robot can be calculated as little less than USD 380, which is around 16 times cheaper than the fourth generation NAO having a price tag of USD 6500. So it is clear that the Tinku is much more affordable than NAO [55].

Table 2. Table of components used.

S. No.	Item
1	Arduino Mega 1280
2	Udoo quad
3	DC motor driver (L298)
4	Logitech webcam c-270
5	6 Ultrasonic distance sensor
6	11.1v Lipo battery
7	Voltage regulators
8	Touch screen (7″)
9	T plug male connector
10	Servo motor driver
11	Sensor connector PCB
12	3 servo motors
13	2 DC motors
14	Robot's body
15	Miscellaneous

6.1.3. Robustness

Tinku has two versions, version 1 and version 2. Figure 7 shows the two versions of the robot. The main difference between these two versions is the design and the material used for the robot's body. Version 1 was significant compared to version 2, made out of acrylic plastic. Version 2 is made out of mainly wood, and the base plate is made of steel. The material cost of version 2 (wood) is less than version 1 (acrylic plastic) due to its smaller size. The manufacturing cost of version 2 is higher because of its design and steel used as steel is difficult to work on as compared to acrylic plastic. Due to these two contradictory reasons, the overall cost of the robot's body in the two versions has a negligible difference.



(**a**) Version 1 (**b**) Version 2

Figure 7. Two versions of Tinku.

So the reason behind designing a new robot was the structural strength and robustness. Version 2 uses steel as its chassis, which improves its strength and makes it rigid. The distance between the wheels is also reduced, which improves the tension in the track belt, and results in better mobility and less slippage. So, as a result, the overall robustness is improved by redesigning the robot. Additionally, version 2 looks cuter and more friendly than version 1, which complements the robot's friendly look.

6.2. Challenge 2: Software

Challenge number 2 is not as straightforward as challenge number 1. Some of the problems stated in challenge number 1 fall under software problems like making the robot social. Additionally, like the hardware problem, let us break challenge number 2 into smaller parts.

- Make robot social.
- Design lesson for daily tasks.

6.2.1. Make Robot Social

For a robot to be social, it needs to be engaged with by humans. The most prominent way of engaging with humans is communication. Tinku uses three dimensions of communication.

1. Visual communication.
2. Body language.
3. Sound effect.

Visual communication is implemented using the screen, which also doubles as its head and face. The body language is implemented using the three servo neck, which incorporates the head gesture and the wheels, complementing the body language, emotions, and expressions. The third dimension is the sound effect. Tinku cannot talk, whatever it does is non-verbal. The sound effect increases the depth of the expression and makes emotions as accurate as possible. The emotions incorporated in Tinku are

- Happy.
- Sad.
- Anger.
- Excitement.
- Sleepy.

6.2.2. Design Lesson for Day to Day Tasks

The task for which the lessons are designed is brushing. Tinku uses digital media to teach brushing to a student. The task is divided into three lessons. The teaching module in all three lessons is perfectly implemented and working as expected. The base method for teaching an intellectually impaired student uses flashcards to demonstrate the steps involved in the task. Tinku uses digital media like images, gifs, and videos to demonstrate the same steps more effectively and in an engaging way.

6.2.3. Implementation of a Deep Learning Algorithm on Development Board

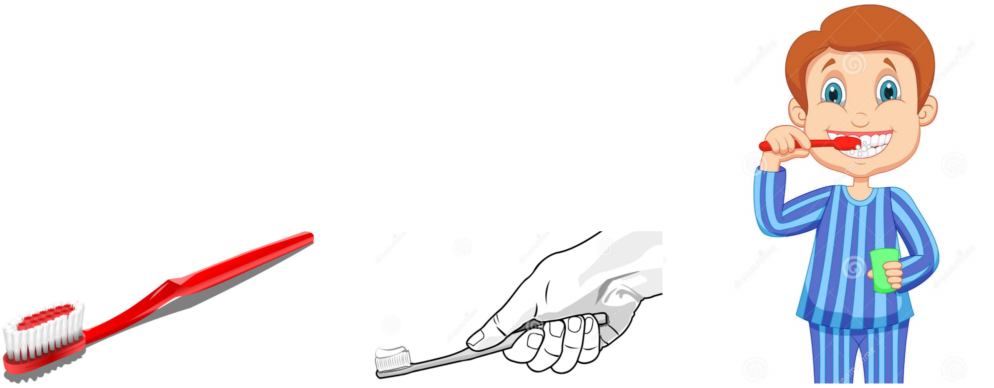
The object detection algorithms were tested on the development board for their latency. We have compared the Tiny-YOLO and TF-lite models with the SSD network. It is shown that the TF-lite model has a better detection speed and mAP value, so we decided to use TF-lite for our research purposes. A comparison is being done in Table 3 for used object detection techniques.

TF-lite is a wrapper function, which facilitates to use a deep learning model with reduced parameters for the same purpose. It provides a very light model with the same level of efficiency for detection.

Table 3. Comparison of two object detection techniques; SSD and YOLO, on development board.

Techniques	Detection Speed (fps)	Reported mAP
TF-Lite (SSD Mobilenet)	2.5	82%
Tiny- YOLO	0.7	23%

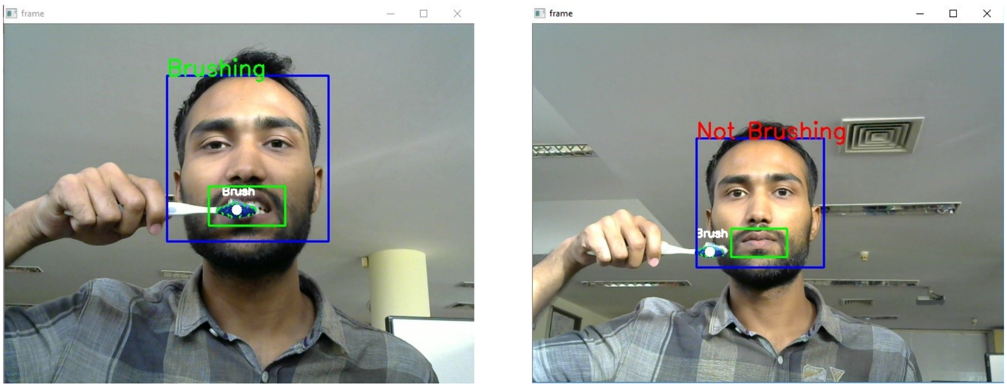
Figure 8 shows the images used in different lessons to teach about brushing. Figure 8a shows the image used to teach what a brush looks like. Figure 8b shows the image used to teach how to hold the brush, and Figure 8c shows the image used to teach how to brush.



(**a**) Lesson 1 (**b**) Lesson 2 (**c**) Lesson 3

Figure 8. Digital media sample, used to teach in the different lessons.

The testing methods used in the lessons are unique too. Lesson 1 gives feedback in the form of emotions and also, at the same time, encourages the student to perform better. Lesson 3 uses the computer vision technique and determines whether the student is brushing or not. Figure 9 shows the result of the test module in lesson 3. Figure 9a shows the output of the test when the student is brushing, and Figure 9b shows the output of the test when the student is not brushing.



(**a**) Brushing (**b**) Not brushing

Figure 9. Test results of lesson 3.

7. Conclusions

In this paper, we have successfully tested our proposed robotic framework for supporting clinical trials for recovery with autism spectral disorder. We have solved the problem of a high cost while implementing sophisticated setups such as a robot for general usage. Tinku is a cost-effective setup for deploying it to a typical household setup. The proposed framework is validated by psychology and robotics experts and tested for its acceptance among young children. The lessons are based on mathematical approaches, making them more explainable for medical applications.

It can be extended by testing it on different test cases and doing extensive clinical trials. The psychological aspect of testing makes this approach more robust for use in case of autism.

Author Contributions: Conceptualization, A.S. and A.M.R.; Methodology, A.S. and K.R.; Software, S.V.; Formal analysis, S.V.; Investigation, K.R.; Data curation, A.S.; Writing—original draft, A.S., T.K. and A.M.R.; Writing—review & editing, K.R., T.K. and A.M.R.; Visualization, S.V. All authors have read and agreed to the published version of the manuscript.

Funding: No funding information available.

Data Availability Statement: No associated data available.

Acknowledgments: This project has been supported by Ministry of Education, India.

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

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolution neural network
DNN	Deep neural network
CV	Computer vision
ML	Machine learning
NLP	Natural language processing
STEM	Science, technology, engineering, and mathematics
AI	Artificial intelligence

References

1. Aleem, S.; Kumar, T.; Little, S.; Bendechache, M.; Brennan, R.; McGuinness, K. Random data augmentation based enhancement: A generalized enhancement approach for medical datasets. *arXiv* **2022**, arXiv:2210.00824.
2. Kumar, T.; Park, J.; Ali, M.; Uddin, A.; Bae, S. Class Specific Autoencoders Enhance Sample Diversity. *J. Broadcast Eng.* **2021**, *26*, 844–854.
3. Khan, W.; Raj, K.; Kumar, T.; Roy, A.; Luo, B. Introducing urdu digits dataset with demonstration of an efficient and robust noisy decoder-based pseudo example generator. *Symmetry* **2022**, *14*, 1976. [CrossRef]
4. Ch, io, A.; Gui, G.; Kumar, T.; Ullah, I.; Ranjbarzadeh, R.; Roy, A.; Hussain, A.; Shen, Y. Precise Single-stage Detector. *arXiv* **2022**, arXiv:2210.04252.2022.
5. Roy, A.; Bose, R.; Bhaduri, J. A fast accurate fine-grain object detection model based on YOLOv4 deep neural network. *Neural Comput. Appl.* **2022**, *34*, 3895–3921. [CrossRef]
6. Naude, J.; Joubert, D. The Aerial Elephant Dataset: A New Public Benchmark for Aerial Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 16–20 June 2019; pp. 48–55.
7. Kim, Y.; Park, J.; Jang, Y.; Ali, M.; Oh, T.; Bae, S. Distilling Global and Local Logits with Densely Connected Relations. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 6290–6300.
8. Tran, L., Ali, M.; Bae, S. A Feature Fusion Based Indicator for Training-Free Neural Architecture Search. *IEEE Access* **2021**, *9*, 133914–133923. [CrossRef]
9. Ali, M.; Iqbal, T.; Lee, K.; Muqeet, A.; Lee, S.; Kim, L.; Bae, S. ERDNN: Error-resilient deep neural networks with a new error correction layer and piece-wise rectified linear unit. *IEEE Access* **2020**, *8*, 158702–158711. [CrossRef]
10. Khan, W.; Turab, M.; Ahmad, W.; Ahmad, S.; Kumar, K.; Luo, B. Data Dimension Reduction makes ML Algorithms efficient. *arXiv* **2022**, arXiv:2211.09392.
11. Kumar, T.; Park, J.; Bae, S. Intra-Class Random Erasing (ICRE) augmentation for audio classification. In Proceedings of the Korean Society of Broadcast Engineers Conference, Las Vegas, NV, USA, 23–27 April 2022; pp. 244–247.
12. Park, J.; Kumar, T.; Bae, S. Search for optimal data augmentation policy for environmental sound classification with deep neural networks. *J. Broadcast Eng.* **2020**, *25*, 854–860.
13. Turab, M.; Kumar, T.; Bendechache, M.; Saber, T. Investigating multi-feature selection and ensembling for audio classification. *arXiv* **2022**, arXiv:2206.07511.
14. Park, J.; Kumar, T.; Bae, S. Search of an Optimal Sound Augmentation Policy for Environmental Sound Classification with Deep Neural Networks. 2020; pp. 18–21. Available online: https://koreascience.kr/article/JAKO202001955917251.do (accessed on 16 November 2022).
15. Sarwar, S.; Turab, M.; Channa, D.; Chandio, A.; Sohu, M.; Kumar, V. Advanced Audio Aid for Blind People. *arXiv* **2022**, arXiv:2212.00004.
16. Singh, A.; Ranjbarzadeh, R.; Raj, K.; Kumar, T.; Roy, A. Understanding EEG signals for subject-wise Definition of Armoni Activities. *arXiv* **2023**, arXiv:2301.00948.

17. Ullah, I.; Khan, S.; Imran, M.; Lee, Y. RweetMiner: Automatic identification and categorization of help requests on twitter during disasters. *Expert Syst. Appl.* **2021**, *176*, 114787. [CrossRef]
18. Kowsari, K.; Meimandi, K.J.; Heidarysafa, M.; Mendu, S.; Barnes, L.; Brown, D. Text classification algorithms: A survey. *Information* **2019**, *10*, 150. [CrossRef]
19. Jamil, S.; Abbas, M.S.; Roy, A.M. Distinguishing Malicious Drones Using Vision Transformer. *AI* **2022**, *3*, 260–273. [CrossRef]
20. Roy, A.M.; Bhaduri, J. Real-time growth stage detection model for high degree of occultation using DenseNet-fused YOLOv4. *Comput. Electron. Agric.* **2022**, *193*, 106694. [CrossRef]
21. Roy, A.M.; Bhaduri, J. A deep learning enabled multi-class plant disease detection model based on computer vision. *AI* **2021**, *2*, 413–428. [CrossRef]
22. Roy, A.M. An efficient multi-scale CNN model with intrinsic feature integration for motor imagery EEG subject classification in brain-machine interfaces. *Biomed. Signal Process. Control* **2022**, *74*, 103496. [CrossRef]
23. Roy, A.M. A multi-scale fusion CNN model based on adaptive transfer learning for multi-class MI classification in BCI system. *bioRxiv* **2022**. [CrossRef]
24. Roy, A.M. Adaptive transfer learning-based multiscale feature fused deep convolutional neural network for EEG MI multiclassification in brain–computer interface. *Eng. Appl. Artif. Intell.* **2022**, *116*, 105347. [CrossRef]
25. Bose, R.; Roy, A. *Accurate Deep Learning Sub-Grid Scale Models for Large Eddy Simulations*; Bulletin of the American Physical Society: New York, NY, USA, 2022.
26. Khan, W.; Kumar, T.; Cheng, Z.; Raj, K.; Roy, A.; Luo, B. SQL and NoSQL Databases Software architectures performance analysis and assessments—A Systematic Literature review. *arXiv* **2022**, arXiv:2209.06977.
27. Dillmann, R. Teaching and learning of robot tasks via observation of human performance. *Robot. Auton. Syst.* **2004**, *47*, 109–116. [CrossRef]
28. Sahin, A.; Ayar, M.; Adiguzel, T. STEM Related After-School Program Activities and Associated Outcomes on Student Learning. *Educ. Sci. Theory Pract.* **2014**, *14*, 309–322. [CrossRef]
29. Mubin, O.; Stevens, C.; Shahid, S.; Al Mahmud, A.; Dong, J. A review of the applicability of robots in education. *J. Technol. Educ. Learn.* **2013**, *1*, 13. [CrossRef]
30. Singh, A.; Narula, R.; Rashwan, H.; Abdel-Nasser, M.; Puig, D.; Nandi, G. Efficient deep learning-based semantic mapping approach using monocular vision for resource-limited mobile robots. *Neural Comput. Appl.* **2022**, *34*, 15617–15631. [CrossRef]
31. Kumar, T.; Park, J.; Ali, M.; Uddin, A.; Ko, J.; Bae, S. Binary-classifiers-enabled filters for semi-supervised learning. *IEEE Access* **2021**, *9*, 167663–167673. [CrossRef]
32. Chio, A.; Shen, Y.; Bendechache, M.; Inayat, I.; Kumar, T. AUDD: Audio Urdu digits dataset for automatic audio Urdu digit recognition. *Appl. Sci.* **2021**, *11*, 8842.
33. Singh, A.; Pandey, P.; Nandi, G. Influence of human mindset and societal structure in the spread of technology for Service Robots. In Proceedings of the 2021 IEEE 8th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), Dehradun, India, 11–13 November 2021; pp. 1–6.
34. Belpaeme, T.; Kennedy, J.; Ramachandrran, A.; Scassellati, B.; Tanaka, F. Social robots for education: A review. *Sci. Robot.* **2018**, *3*, eaa5954.
35. Billard, A. Robota: Clever Toy and Educational Tool. *Robot. Auton. Syst.* **2003**, *42*, 259–269. Available online: http://robota.epfl.ch (accessed on 16 November 2022). [CrossRef]
36. Ricks, D.; Colton, M. Trends and considerations in robot-assisted autism therapy. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–8 May 2010; pp. 4354–4359.
37. Breazeal, C.; Kidd, C.; Thomaz, A.; Hoffman, G.; Berlin, M. Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. In Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS 2005), Edmonton, AB, Canada, 2–6 August 2005.
38. Fong, T.; Nourbakhsh, I.; Dautenhahn, K. A survey of socially interactive robots. *Robot. Auton. Syst.* **2003**, *42*, 143–166. [CrossRef]
39. Bar-Cohen, Y.; Breazeal, C. Biologically inspired intelligent robots. In Proceedings of the Smart Structures and Materials 2003: Electroactive Polymer Actuators and Devices (EAPAD), San Diego, CA, USA, 3–6 March 2003; Volume 5051, pp. 14–20.
40. Kidd, C.; Breazeal, C. Effect of a robot on user perceptions. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), Sendai, Japan, 28 September–2 October 2004; Volume 4, pp. 3559–3564.
41. Breazeal, C. Toward sociable robots. *Robot. Auton. Syst.* **2003**, *42*, 167–175. [CrossRef]
42. Maleki, F.; Farhoudi, Z. Making Humanoid Robots More Acceptable Based on the Study of Robot Characters in Animation. *IAES Int. J. Robot. Autom.* **2015**, *4*, 63. [CrossRef]
43. School, T. Topcliffe Primary School. 2021. Available online: http://www.topcliffe.academy/nao-robots (accessed on 16 November 2022).
44. Lite, T. TensorFlow Lite. 2021. Available online: https://tensorflow.org/lite (accessed on 16 November 2022).
45. Phadtare, M.; Choudhari, V.; Pedram, R.; Vartak, S. Comparison between YOLO and SSD Mobile Net for Object Detection in a Surveillance Drone. *Int. J. Sci. Res. Eng. Man* **2021**, *5*, 1–5.

46. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 24–27 June 2014; pp. 580–587.
47. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
48. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 2969239–2969250. [CrossRef]
49. Adarsh, P.; Rathi, P.; Kumar, M. YOLO v3-Tiny: Object Detection and Recognition using one stage improved model. In Proceedings of the 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 6–7 March 2020; pp. 687–694.
50. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.; Berg, A. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
51. Ning, C.; Zhou, H.; Song, Y.; Tang, J. Inception single shot multibox detector for object detection. In Proceedings of the 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Hong Kong, China, 10–14 July 2017; pp. 549–554.
52. Roy, A.; Bhaduri, J.; Kumar, T.; Raj, K. WilDect-YOLO: An efficient and robust computer vision-based accurate object localization model for automated endangered wildlife detection. *Ecol. Inform.* **2022**, 101919. [CrossRef]
53. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
54. Ding, S.; Long, F.; Fan, H.; Liu, L.; Wang, Y. A novel YOLOv3-tiny network for unmanned airship obstacle detection. In Proceedings of the 2019 IEEE 8th Data Driven Control and Learning Systems Conference (DDCLS), Dali, China, 24–27 May 2019; pp. 277–281.
55. RobotLAB Group. NAO Version Six Price. 1981. Available online: https://www.robotlab.com/store/nao-power-v6-standard-edition (16 November 2022).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.