Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Real-Time Performance Evaluation of Household Object Detection, Recognition and Grasping Using Firebird V Robot

Smita Gour[1],*, Pushpa B Patil[2]
[1]*Department of Computer Science and Engineering, Basaveshwar Engineering College, Bagalkot-587102, Karnataka, India*
[2]*Department of Computer Science and Engineering (Data Science), BLDEA's V P Dr P G Halakatti College of Engineering & Technology, Vijayapur-586102, Karnataka, India*
1 smita.gour@gmail.com; 2 pushpapatil2008@gmail.com;*
** corresponding author*

ARTICLEINFO

Received: 29 Dec 2024

Revised: 12 Feb 2025

Accepted: 27 Feb 2025

ABSTRACT

Real-time object detection, recognition, and grasping are essential capabilities for autonomous robotic systems operating in dynamic environments. In this paper, we present a comprehensive performance evaluation of these functionalities using the Firebird V Robot, a versatile platform renowned for its agility and robustness. Leveraging state-of-the-art computer vision algorithms, we assess the system's ability to detect objects in real-time, recognize their identities, and grasp them reliably under varying environmental conditions. Our experimental framework encompasses a series of tests conducted in both simulated scenarios, enabling a thorough analysis of the Firebird V Robot's performance. Through meticulous experimentation and analysis, we provide valuable insights into the practical viability of deploying these functionalities in real-world applications. Our findings such as detection and recognition accuracy (92.2%), grasp success rate (0.8), total execution time (30sec to 60sec) and efficiency shed light on the strengths and limitations of the methods being evaluated and Firebird V Robot as a platform for real-time object manipulation. By evaluating the performance of these critical functionalities, we contribute to the advancement of intelligent robotic systems capable of seamlessly integrating into various domains, from industrial automation to service robotics and beyond.

Keywords: Object detection, Object recognition, Grasping, Firebird V Robot, Evaluation.

1. INTRODUCTION

In recent years, advancements in robotics and computer vision have propelled the development of intelligent systems capable of real-time object detection [1], recognition [2], and grasping [3]. These capabilities hold immense promise for various applications, ranging from industrial automation [4] to service robotics [5] and beyond.

The ability to detect objects in real-time is a fundamental requirement for autonomous robotic systems operating in dynamic environments. The methods involved in object detection such as [6] where the classification method used is Support Vector Machine (SVM) and as input to this system, RGB and depth images are used. Different segmentation techniques have been applied to each kind of object. Other recent techniques used for object detection are pre trained YOLOV4 [7] and Deep Convolutional Neural Network (CNN) model, called KSSnet [8] is developed for object detection based on CNN Alexnet using transfer learning approach. Object recognition further enhances this capability by enabling robots to differentiate between various objects and make informed decisions based on their identities. Approaches such as You Only Look Once (YOLO) [9], Faster R-CNN [10], and Single Shot MultiBox Detector (SSD) [11] have demonstrated remarkable performance in detecting and recognizing objects in various setting.

Moreover, the capability of grasping objects reliably is essential for executing tasks ranging from pick-and-place operations in manufacturing settings to assisting individuals with activities of daily living. The field of real-time grasping has seen notable progress with the development of deep learning-based methods for planning and executing robust grasps. Studies such as Dex-Net 2.0 [12], Learning Hand-Eye Coordination [13], and Supersizing

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Self-Supervision [14] have showcased the efficacy of deep learning in learning grasping policies from large-scale data. Recent research has focused on integrating real-time perception and grasping capabilities on robotics platforms. Techniques such as adapting deep visuomotor representations [15], probabilistic grasp detection with convolutional neural networks [16], and learning synergies between pushing and grasping [17] have shown promise in improving the efficiency and reliability of robotic manipulation tasks. To execute all these tasks there exists number of robotic systems both virtually and physically such as service and interactive robots. Among them MOnarCH robot (mbot) [18], originally designed to interact with children inside hospitals and to participate at domestic robot competition, ASIMO Robot [19] is designed for domestic applications, where it needs to perceive the environment, act accordingly and complete one or a few domestic tasks. Lastly multi-purpose domestic robots are usually humanoid robots [19]. For example, NAO and Pepper result in more active interactions with human users as they appear human-like. Among the plethora of robots designed to execute such tasks, the Firebird V Robot [20] stands out for its versatility, agility, and robustness. Throughout this paper, we address key challenges encountered in deploying object detection, recognition, and grasping algorithms on the Firebird V Robot. Furthermore, a set of innovative benchmarks and an automatic performance evaluation system are proposed in [19] and used to evaluate the performance of the developed functionalities.

We provide insights into the performance of these algorithms concerning accuracy, speed, and robustness, shedding light on their practical viability in real-world scenarios. Our experimental setup involves a series of tests conducted in simulated environments, showcasing the Firebird V Robot's ability to perceive and interact with objects in real-time leveraging some issues and challenges.

Evaluating household object recognition using the Firebird V robot involves multiple challenges, ranging from sensor limitations to real-world variability. Below are some key issues and challenges categorized into different aspects:

Hardware & Sensor Limitations – Low-resolution cameras, depth sensor noise, and limited onboard processing can affect performance.

Algorithmic Challenges – Handling occlusions, class imbalance, and poor generalization to unseen objects makes recognition difficult.

Real-Time Performance Issues – High inference times, network delays, and multi-tasking overhead can slow down recognition and response.

Environmental Variability – Changes in lighting, cluttered backgrounds, and reflective surfaces impact object detection accuracy.

Dataset & Training Issues – Limited RGB-D datasets, annotation errors, and domain adaptation problems hinder model robustness.

Uncertainty & Reliability – Ensuring the model quantifies confidence using **Dirichlet-based EDL** is crucial for avoiding misclassifications.

Grasp Planning Challenges – Errors in object pose estimation, slippage, and handling dynamic objects affect successful grasp execution.

Evaluation & Benchmarking Issues – Lack of standardized metrics, the gap between simulation and real-world performance, and experiment repeatability make evaluations inconsistent.

Addressing some of these issues the methodology for real-time performance evaluation of household object detection, recognition and grasping has been proposed.

2. PROPOSED METHODOLOGY

The block diagram showed in Fig. 1 represents the proposed process for performance evaluation of real-time household object detection, recognition, and grasping methods using the Firebird V Robot. It involves a structured pipeline to ensure accurate and efficient robotic manipulation. The process begins with data acquisition, where sensors and cameras on Firebird V Robot capture real-time RGB and depth image of household objects. The image

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

is sent to the detection and recognition system through the XBee wireless communication module. The object detection module YOLOv8n (You Only Look Once) then identifies objects, followed by an object recognition step that classifies detected objects using learning-based models and some feature extraction techniques. Once an object is recognized, the corresponding signal is generated and the grasp planning module computes the optimal grasping points based on object shape, size, and orientation. The Firebird V Robot then executes the grasping action based on the planned strategy. Finally, the system undergoes performance evaluation, measuring key metrics such as detection accuracy, recognition speed, grasp success rate, and real-time feasibility. This structured approach ensures efficient and reliable household object manipulation, making it suitable for robotic automation in domestic and industrial environments. The details of accomplishment of these tasks are given in further sections.

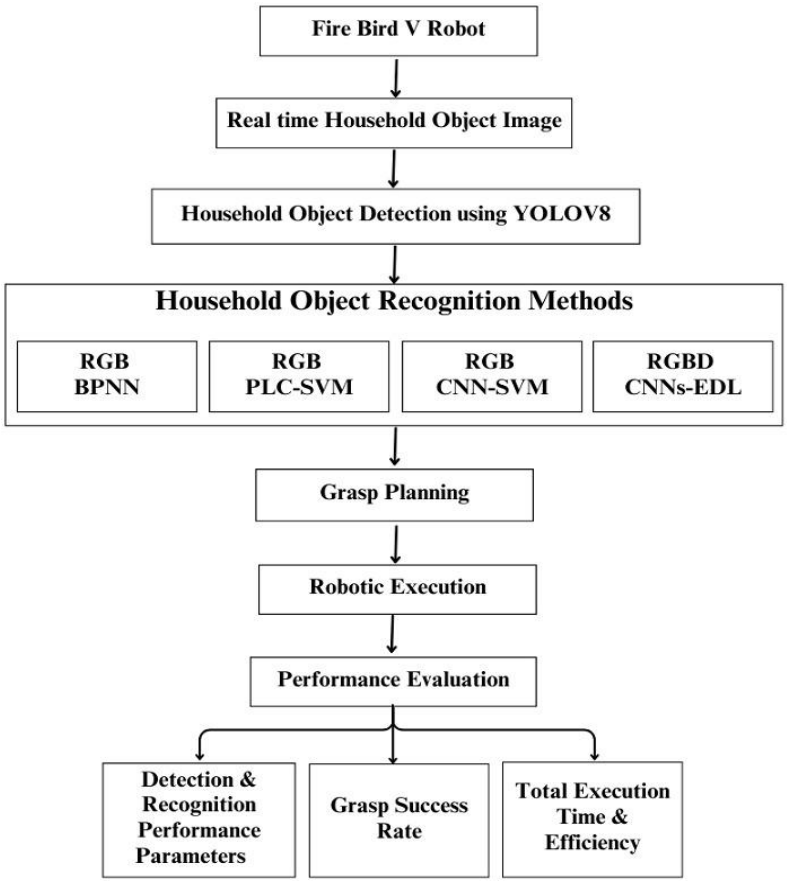

Figure 1. Proposed Evaluation Process

2.1 FireBird V Robot

The Fire Bird V robot is used as a real time system to evaluate some of the household object detection, recognition and grasping methods. The Fire Bird V robot showed in Fig. 2 is the fifth generation in the Fire Bird series [20]. The first two versions were developed for the Embedded Real-Time Systems Lab at the Department of Computer Science and Engineering, IIT Bombay. Starting with version 3, these platforms were made commercially available.

The Fire Bird V supports ATMEGA2560 (AVR), P89V51RD2 (8051), and LPC2148 (ARM7) microcontroller adapter boards, making it a highly versatile platform. Additionally, users can integrate custom-designed microcontroller adapter boards to suit specific needs. The details of its hardware and programming can be found in [21, 22].

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Figure 2: FireBird– V Robot

2.1.1 Interfacing Firebird V Robot

The main parts of Firebird V Robots used in this work are external camera, gripper, zigbee wireless communication module, inbuilt obstacle detection sensors and white line following sensors. It has the facility to fit any suitable camera shown in Fig. 3a, zigbee module shown in Fig 3b and grippers shown in Fig. 3c. After fitting these it looks like in Fig. 3d.

(a) (b) (c) (d)

Figure 3: (a) Camera pod provided with Firebird V robot (b) Gripper provided with Firebird V robot (c) XBee Wireless Communication module (d) Assembled Firebird V robot

Two low torque servo motors are used in the camera pod assembly to precisely regulate the tilt and pan angles. The Firebird V robot's top side can easily accommodate this camera pod. The objects are retrieved using the external camera. Around 180^{o}, it will turn from left to right and vice versa. The gripper is designed for robotics applications and is lightweight. It may cling to objects up to 40 mm in diameter. A gripper allows one to grasp, tighten, handle, and release an object, just like a hand does.

Gripper is fully built and functional, including the acrylic parts. The mechanical link design allows the gripper jaws to move almost parallel to each other when gripping. The gripper comes with two high-end dual bearing NRS-585 servo motors. One servo motor is attached for gripping, and the second servo motor is attached for up-and-down motion.

XBee modules are used at both ends to establish wireless communication between the detection and recognition system and Firebird V robot, eliminating the need for a wired USB-to-Serial connection. XBee modules use Zigbee (IEEE 802.15.4) protocol, which provides a low-power, long-range, and reliable wireless communication method.

With this arrangement the samples of real time household objects captured by a robot during testing of algorithms is shown in Fig. 4. The Programming of this robot is discussed in next section.

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Figure 4: Samples captured by Firebird V

2.1.2 Programming Firebird V Robot

Programming the Firebird V robot with ATmega2560 involves writing embedded C code, compiling it using AVR-GCC, and uploading it via a programmer like AVRISP mkII or USBasp. First, we have to install WinAVR or Atmel Studio for coding and compilation. The Firebird V robot's firmware typically includes functions for motor control, sensor interfacing (IR, ultrasonic, etc.), and serial communication (UART, SPI, I2C). The code is structured with initialization functions (setting up GPIOs, timers, PWM, and ADC), followed by main logic controlling the robot's movements or responses to sensor inputs. The compiled .hex file is then uploaded using AVRDUDE or Atmel Studio's programming interface. Debugging can be done using serial print statements (UART) or onboard LEDs for status indications. Proper power supply (battery or USB) and ensuring correct baud rates for communication (typically 9600 bps) are crucial for smooth operation.

2.2 Household Object detection

The household image or video captured by Firebird V is processed by OpenCV to detect object in an image using YOLOv8 [23]. It is a powerful, real-time object detection model that can efficiently identify household objects in images, videos, or live camera feeds. With pre-trained weights, it can detect objects out-of-the-box, and for our system it is trained with 50 samples of each of 10 different household objects like bottle, spoon, fork, toothbrush, cup, apple, phone, scissor, alarm and knife. The model's speed and accuracy make it ideal for smart home automation, robotics, and AI-driven assistance applications.

2.3 Household Object Recognition Algorithms

Many methods of object recognition systems exist in literature. We have proposed four different methods on which we want to perform a suggested evaluation process. They are discussed in further section.

2.3.1 Back Propogation Neural Network based Recognition method (RGB-BPNN)[24]

Here a traditional machine learning method for identifying common objects is implemented. This method employs a feature-based methodology that aims for recognition in a variety of situations. To identify the items, the appropriate image processing methods are used. These methods include extracting shape and texture features from the object images, removing the shadow that separates the object from its shadow, and developing descriptors that partially get around the challenges of affine transformations. The "Back Propagation Neural Network" (BPNN) is trained with Washington RGB images and with some real images of household objects. It classifies objects into their appropriate classes based on this past knowledge of descriptors. With the help of 38 potent combined features of shape and texture, and BPNN properties, the system produced the desired outcome on standard Washington RGB images of objects.

2.3.2 Support Vector Machine based Recognition method (RGB-PLC-SVM) [25]

With the use of the Point Cloud Library (PCL), the suggested system seeks to create and construct an object recognition system. A three-stage approach shown in Fig. 5 is used to solve the object recognition problem. The object image is subjected to segmentation techniques in the first step of PCL, which prepares it for feature extraction. In the second step, appropriate shape-based characteristics that can distinguish between different object types are extracted from a segmented image. The final step entails utilizing Support Vector Machines (SVM) to

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

classify or recognize the object of the specific category. Using the Point Cloud Library (PCL) and Support Vector Machine (SVM) as a classifier, the system produced the anticipated outcomes. For ten distinct categories of home objects from Washington RGBD dataset, the system has provided an accuracy of 94%. Here in this article we present an evaluation of the same method in real time using Fire birt V robot.

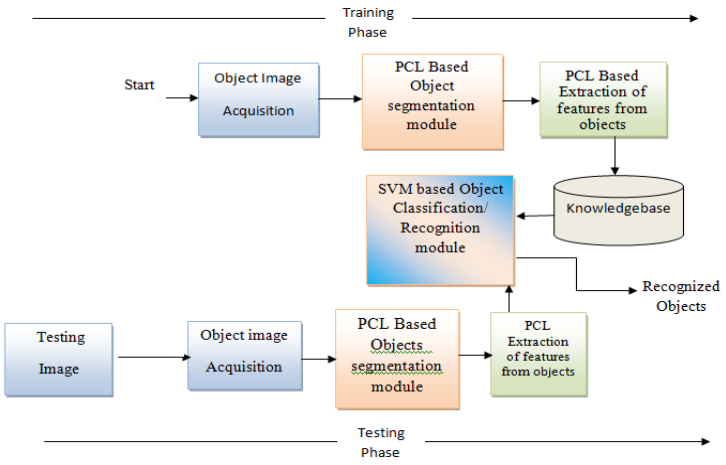

Figure 5: Proposed SVM based Recognition System

2.3.3 Convolutional Neural Network and Support Vector Machine based System (RGB-CNN-SVM) [26]

This work introduces a comprehensive object recognition system designed to empower robots to perceive and recognize diverse objects within their surroundings. Leveraging the capabilities of Convolutional Neural Networks (CNNs), the proposed system aims to address the intricacies associated with extracting robust and accurate features of household objects. Support Vector Machines (SVMs) are combined with CNN to classify household objects such as cup, book, plate, spoon, chair and like.

The proposed methodology shown in Fig. 6 consists of three phases. These are 1) Building and training CNN model for feature extraction. 2) building and training SVM model using these features to classify household objects. 3) Testing phase. CNN Model will be trained using RGBD dataset and knowledge base is created which will be fed to SVM model to train it to recognize objects.

This deep learning application was developed using the Tensorflow framework. Data augmentation and parameter tuning have been used to boost the system's performance. It exhibited an accuracy of 92% with RGB-D standard dataset.

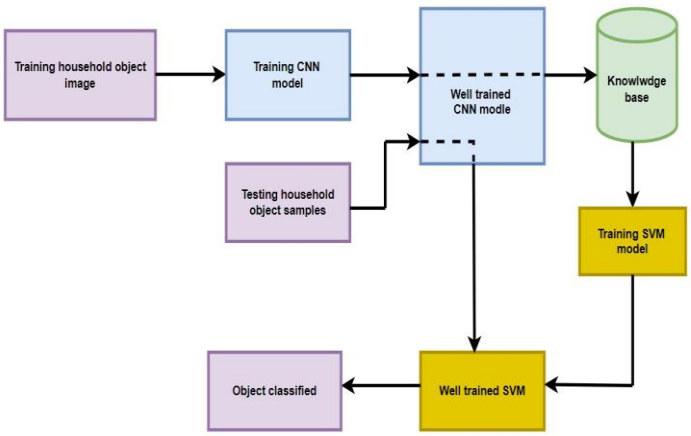

Figure 6: Proposed Hybrid CNN-SVM based Recognition System

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

2.3.4 *Dirichlet-Evidential Deep Learning based System (RGBD-CNN-EDL) [27]*

Traditional deep learning models struggle with sensor noise, occlusions, and overconfident misclassifications. To address this, we propose an Evidential Multimodal Deep Learning (EMDL) framework shown in Fig. 7, integrating Evidential Deep Learning (EDL) with CNN (Convolutional Neural Network) and attention based feature fusion. Our model extracts features using CNNs for RGB and depth, and then fuses them through a cross-attention mechanism, allowing adaptive weighting of modalities based on uncertainty. Instead of softmax classifiers, Dirichlet-based evidential output layer has been used.

It quantifies both classification confidence and epistemic uncertainty, improving robustness. Evaluations on the Washington RGB-D dataset demonstrate superior performance in classification accuracy, noise handling, and domain generalization compared to baseline models. Accuracy of 92.2% is reached with this novel approach considering 10-fold cross validation method. By enhancing uncertainty-aware decision-making, our approach ensures safer and more reliable robotic perception, making it suitable for real-world applications like grasping, manipulation, and autonomous navigation.

Here in this article we present an evaluation of all the above four methods in real time using Fire birt V robot and the details are presented in the ensuing sections.

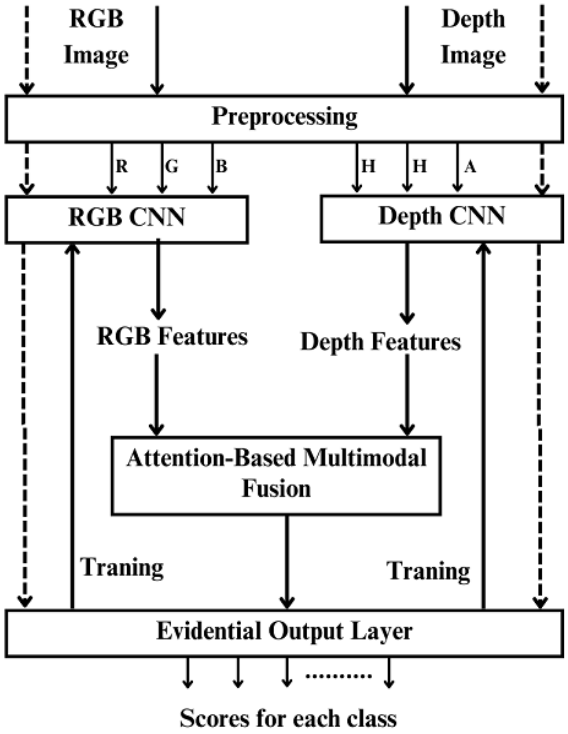

Figure 7: Proposed EDL based Recognition System

2.4 Grasp Planning

After recognizing objects the robot will receive signal for grasp planning. The grasp planning in Firebird V Robot is simple using servo motors. The Firebird V robot can be equipped with a servo-driven gripper, such as the NRS-585 servo motor, to perform grasping tasks. The NRS-585 is a high-torque servo motor commonly used for robotic applications where precise positioning and controlled force are required.

The NRS-585 servo motor gripper operates by setting a PWM (Pulse Width Modulation) signal to control the gripper's open and close positions. The Firebird V's Atmega2560 microcontroller sends PWM signals to the NRS-585 to set these positions. The servo angle defines how much the gripper opens or closes:

- Open Position (90° to 120°) → Gripper is fully open.

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

- Grasp Position (30° to 60°) → Gripper closes around the object.
- Fully Closed Position (0° to 20°) → Used for thin objects like paper.

2.5 Robot Execution

After completing grasp planning, the Firebird V robot executes the grasping task by precisely controlling its robotic arm and NRS-585 servo motor gripper. The process begins with the arm moving towards the detected object's location using inverse kinematics and pre-planned trajectories, ensuring smooth and collision-free motion. Once aligned, the gripper closes securely around the object, adjusting its grip strength to prevent slippage or excessive force. The robotic arm then lifts the object smoothly, maintaining stability while following a controlled trajectory. Next, the robot transports the object to the target location using predefined motion planning algorithms, avoiding obstacles if necessary. Upon reaching the destination, the gripper carefully opens to release the object, ensuring a safe and precise placement. Finally, the robotic arm returns to its home position, preparing for the next grasping task. Throughout execution, real-time feedback from sensors (if available) can enhance accuracy and adaptability.

2.6 Performance Evaluation

The performance evaluation of real-time household object detection, recognition, and grasping using the Firebird V robot focuses on assessing the system's efficiency in identifying and manipulating various household objects. The evaluation was conducted using four different recognition approaches—RGB-BPNN, PLC-SVM, RGB-CNN-SVM, and RGBD-CNN-EDL—to compare their effectiveness in terms of recognition accuracy, grasp success rate, and execution time**.** The detailed experimental setup and performance analysis is discussed in next section.

3. RESULTS

3.1 GUI Building

To interact with the robot we have designed simple GUI as showed in Fig. 8 using the tkinter package library of Python. It provides a robust and platform independent windowing toolkit. We have listed 10 sample household objects in GUI to give input to the system. We also provided two buttons to start and stop. When user clicks on start the robot receives signal to start its task. Once the robot completes its task user will press stop so that robot will move to its initial position.

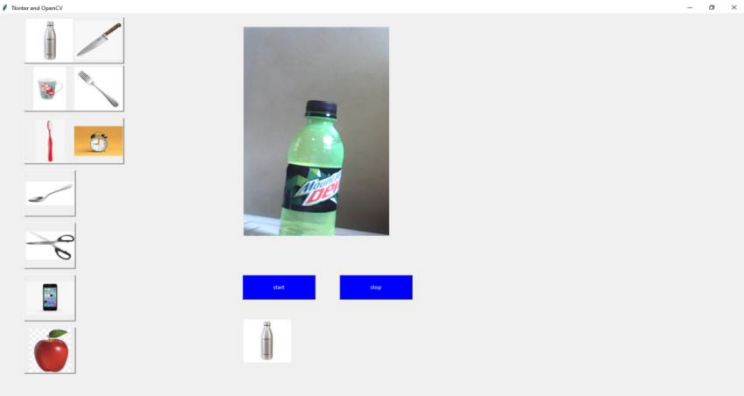

Figure 8: Simple GUI

3.2 Testing Environment

Using a predefined path printed on flex showed in Fig. 9 for evaluation provides a structured and repeatable way to test object detection, recognition, and grasping. This approach is useful for benchmarking performance before deploying the system in real-world environments. This setup ensures a controlled environment for testing and analyzing the robot's performance. The empty red box in a flex shown indicates starting point of robot. The numbered red boxes indicate position of 10 different objects considered in this work. Black lines constitute a predefined path of robot where robot uses its white line sensors to follow this path. Each black point at the middle line is point where the robot stays and recognizes the user given input through GUI.

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

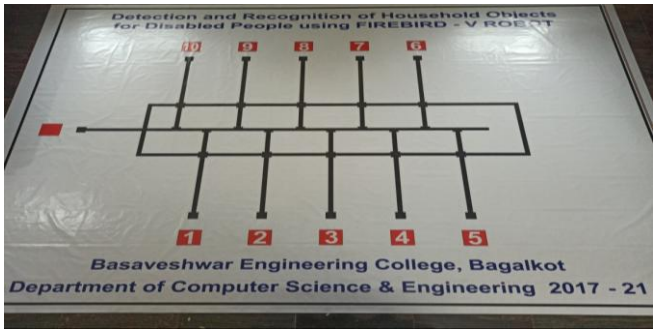

Figure 9: Flex Print of Predefined Path and 10 locations of objects

3.3 Result Analysis

This result analysis aims to evaluate the performance of four different object recognition approaches—RGB-BPNN, PLC-SVM, RGB-CNN-SVM, and RGBD-CNN-EDL—based on key performance metrics such as detection efficiency, recognition efficiency, grasp success rate, and execution time. The evaluation is conducted using 10 different household objects showed in Fig. 10. They are of varying shapes, sizes, and materials to ensure robustness. 50 samples of each of these are collected to conduct object detection. Object detection efficiency measures a model's ability to speedily and accurately identify and localize objects in images or videos. It is evaluated using metrics such as inference time and detection accuracy. Recognition accuracy measures how well each approach classifies objects, while the grasp success rate indicates the system's ability to correctly pick and manipulate objects. Additionally, execution time is analyzed to determine the efficiency of each approach in real-time applications. This analysis provides insights into the effectiveness of different machine learning and deep learning models, highlighting the advantages of multimodal fusion and uncertainty-aware learning in improving object recognition and grasping performance.

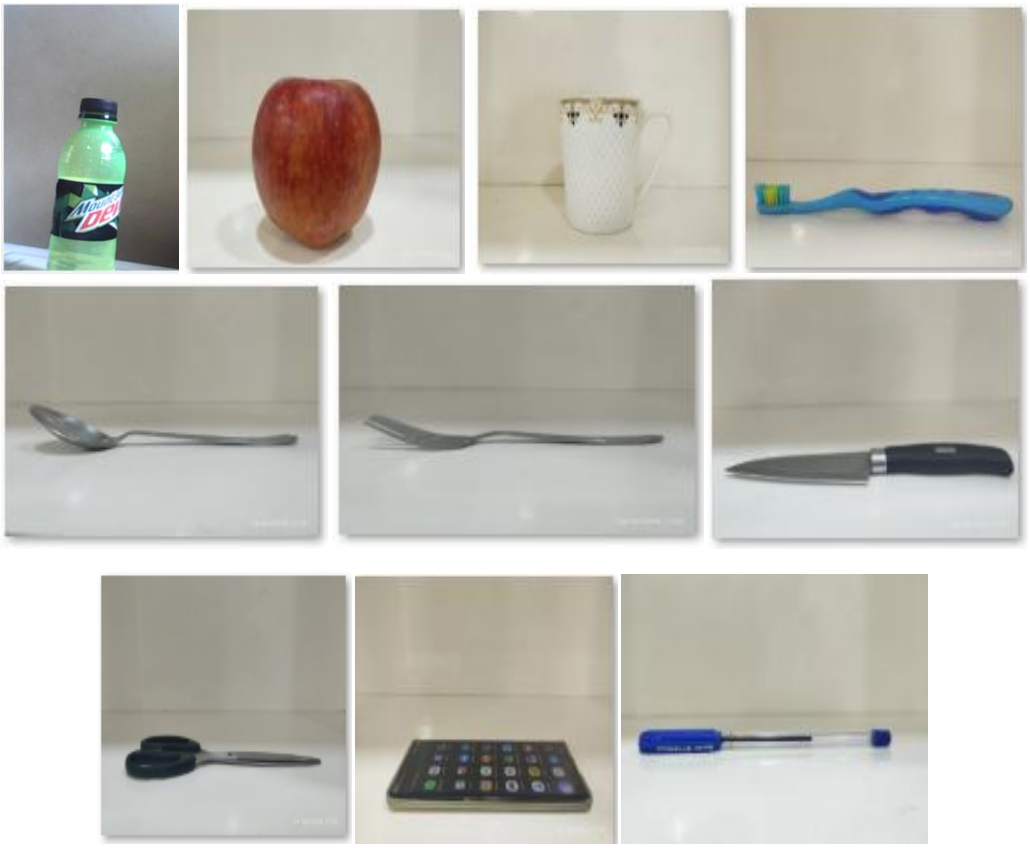

Figure 10: Household objects Considered in this Study

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

3.3.1 Object Detection Efficiency

The detection algorithm used in this study is YOLOv8. In the family of YOLOv8, YOLOv8n (nano) is the most lightweight and efficient model, optimized for low-latency performance on edge devices. It strikes a balance between speed and accuracy, making it well-suited for real-time object detection on CPUs and resource-constrained hardware.

Object detection accuracy measures how well the system identifies objects within an image. This is typically evaluated using Mean Average Precision (mAP), which calculates the average precision across all detected object classes. Precision and recall are key metrics to calculate mAP. They are given by Equation (1) and (2) respectively.

$$\text{Precision } P = \frac{TP}{(TP+FP)} \quad (1)$$

$$\text{Recall } R = \frac{TP}{(TP+FN)} \quad (2)$$

Average precision is given by Equation (3)

$$AP = \sum_{i=1}^{N}(R_i - R_{i-1}) \times P_i \quad (3)$$

Where N is the number of recall levels and P_i is precision at each recall level

The mAP is the mean of the AP values over all classes and is given by Equation (4)

$$mAP = \frac{1}{C}\sum_{c=1}^{c} AP_c \quad (4)$$

Where C is the number of object classes and AP_c is the Average Precision for class c

Here in Fig. 12 a bar chart showing the mean Average Precision (mAP) of YOLOv8n for 10 different household objects at two IoU thresholds such as mAP@50 (IoU 0.5) – a more relaxed evaluation and mAP@50:95 (IoU 0.5 to 0.95) – a stricter evaluation

This visualization highlights how YOLOv8n performs across different object categories, with generally higher accuracy at mAP@50 and a slight drop at mAP@50:95 due to stricter IoU requirements. Fig. 11 shows a sample results produced by YOLOv8n with bounding box and lables.

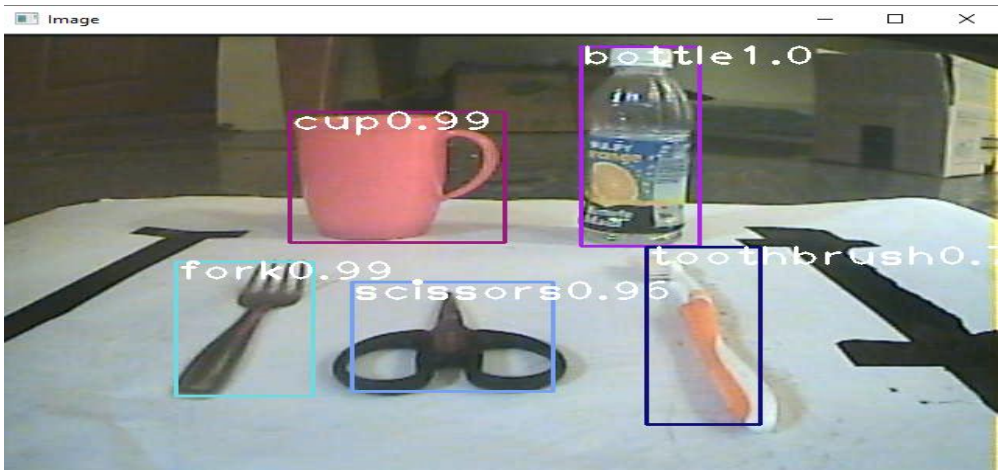

Figure 11: Sample of YOLOv8n Detection Results

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

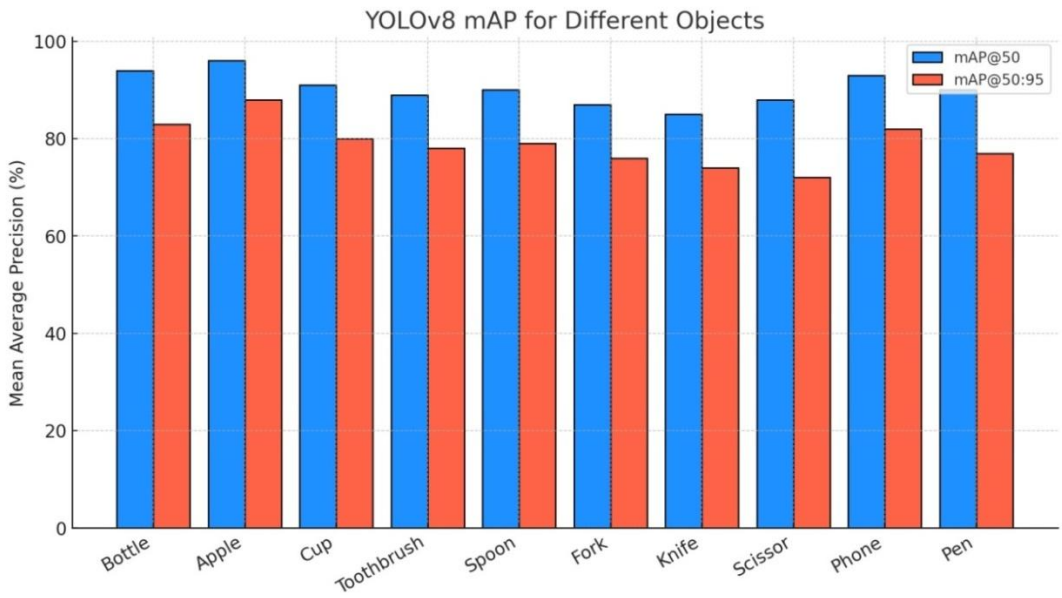

Figure 12: YOLOv8 detection accuracy

The metric considered in this study to measure the speed of a YOLOv8 Detection model is inference time. The inference process consists of three key stages. Preprocessing time involves resizing, normalizing, and preparing the image for the model. Model execution time refers to the duration taken by YOLOv8 to process the image and generate predictions, including bounding boxes, class labels, and confidence scores. Finally, post processing time includes filtering out low-confidence detections, applying Non-Maximum Suppression (NMS) to eliminate duplicate boxes, and formatting the output for further use. The object wise inference time was calculated since the image captured by Firebird V robot contains single object every time. The graph showed in Fig. 13 indicates inference time taken by YOLOv8n for each object to detect on Intel core i5 CPU.

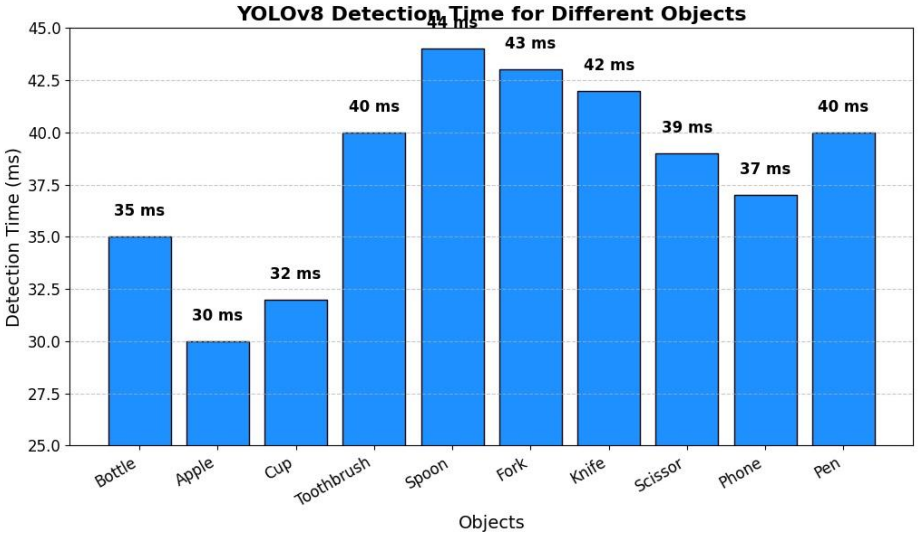

Figure 13: YOLOv8 Detection Time

3.3.2Object Recognition Efficiency

After detecting an object, the system must correctly classify it. The efficiency of household object recognition in robotic vision depends primarily on time (speed of recognition) and accuracy (correct identification of objects). Time efficiency is crucial for real-time applications, as robots need to recognize objects quickly to perform tasks smoothly. Traditional methods such as SIFT and SURF are relatively slow due to their computational complexity,

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

whereas deep learning-based models like YOLO and MobileNet offer faster inference times by using optimized architectures. However, speed often comes at the cost of accuracy—simpler models may run faster but struggle with identifying objects in cluttered or poorly lit environments.

Accuracy in object recognition depends on factors such as training data quality, environmental conditions, and model robustness. Deep learning models trained on large, diverse datasets tend to achieve high accuracy but may require more computational power and memory, leading to longer processing times. Object recognition accuracy is assessed using accuracy metrics given by an equation (5)

$$Accuracy = \frac{Corect\ Predictions}{Total\ Predictions} \times 100 \qquad (5)$$

The efficiency in terms of per object accuracy incurred by the four methods under the study is shown in Fig. 15. Each object is tested 10 times by Firebird V Robot and average accuracy is obtained. This average accuracy is taken as per-object accuracy and the same is depicted in Fig. 14.

Per-object accuracy evaluates an object recognition model's ability to correctly classify individual object categories rather than assessing overall performance across all objects. It quantifies the accuracy of recognizing specific items, such as a bottle, apple, or pen, by determining the percentage of correctly predicted instances for each category. This metric is essential for identifying the model's strengths and weaknesses, as some objects may be recognized with high precision, while others—especially those with complex shapes or similar features—may have lower accuracy. Unlike overall accuracy, which may obscure variations across different object types, per-object accuracy provides a more detailed and balanced assessment of model performance.

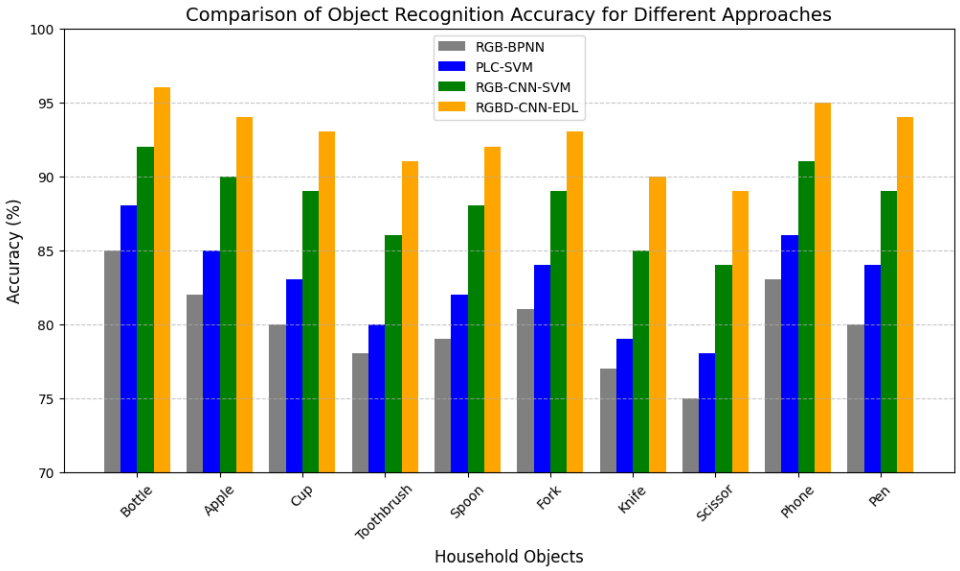

Figure 14: Object Wise Recognition Accuracy of all proposed methods

As shown in Fig. 15 the average accuracy of household object recognition methods varies depending on their feature extraction strategies and classification techniques. RGB-BPNN (Backpropagation Neural Network) has the lowest accuracy, ranging from 77-85%, as it relies on traditional neural networks that struggle with complex object variations. PLC-SVM (Principal Component Analysis + Support Vector Machine) improves accuracy to 79-88% by reducing feature dimensions and utilizing SVM for classification, yet it still lacks deep feature extraction capabilities. RGB-CNN-SVM (Convolutional Neural Network + SVM) further enhances accuracy to 85-92%, leveraging CNN for efficient feature extraction and SVM for precise classification. The highest accuracy, between 89-96%, is achieved by RGBD-CNN-EDL (RGB-D Convolutional Neural Network + Evidential Deep Learning), which integrates both RGB and depth data for enhanced feature learning and uncertainty estimation. This makes it the most reliable approach for robotic vision, offering superior accuracy and robustness in object recognition.

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The recognition time of various household object recognition approaches shown in Fig. 15 differs significantly depending on their underlying methodologies. RGB-BPNN (Backpropagation Neural Network) is the slowest, requiring approximately 120-150 ms per object due to its reliance on traditional neural network training, which demands high computational resources. PLC-SVM (Principal Component Analysis + Support Vector Machine) enhances efficiency by reducing feature dimensions, achieving a moderate speed of 80-100 ms per object. RGB-CNN-SVM (Convolutional Neural Network + SVM) further accelerates recognition, reducing the time to 40-60 ms, as CNN effectively extracts features while SVM efficiently classifies them. The fastest approach, RGBD-CNN-EDL (RGB-D Convolutional Neural Network + Evidential Deep Learning), integrates both RGB and depth data, enhancing feature extraction and reducing uncertainty, achieving 20-40 ms per object. This makes it particularly suitable for real-time robotic vision applications. Ultimately, while traditional methods suffer from higher latency, deep learning-based techniques, especially those incorporating depth information, offer the best balance between speed and accuracy.

3.3.3 Grasp success rate

The grasp success rate evaluates how well the robot successfully picks up objects after detection and recognition. It is computed using the equation (6)

$$Grasp\ Success\ Rate = \frac{Total\ Attempts}{Successful\ Grasps} \times 100 \qquad (6)$$

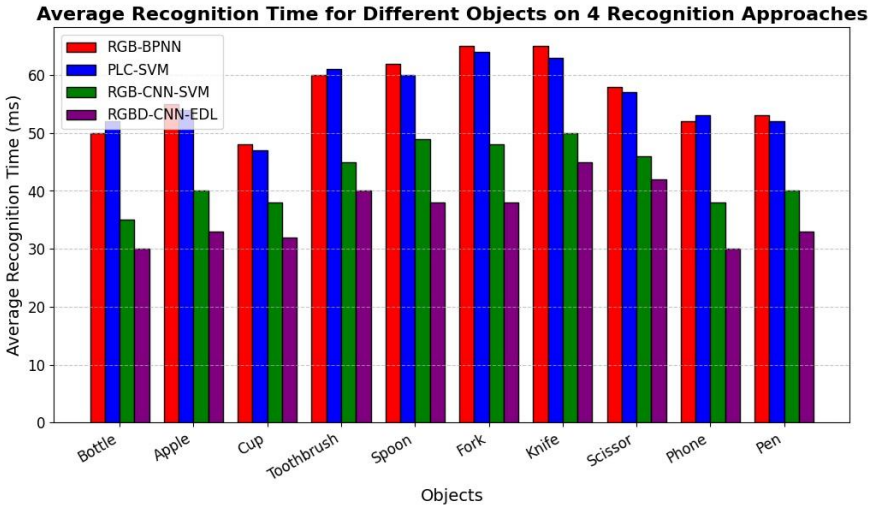

Figure 15: Object Wise Recognition Time of all proposed methods

A successful grasp occurs when the object is securely held, lifted, and placed correctly without slipping or dropping. This metric is influenced by object shape, orientation, and grasp point selection. Algorithms like GGCNN (Generative Grasp CNN) and Dex-Net help in improving grasp planning to maximize this success rate. Firebird V robot uses its gripper to hold, lift and place an object successfully as showed in Fig. 16. Each of ten objects is placed in their place. The recognized objects are attempted 10 times to grasp and the grasp success rate is calculated each time using equation (6). Later the average grasp success rate is obtained for each object and the same is depicted in the graph showed in Fig. 17.

Another metric in measuring grasp efficiency is to determine the average grasp time taken by the Fire Bird V robot. It is crucial to measure the duration from the initiation to the completion of the grasp. This timeframe begins when the robot starts its gripping action and ends once the object is securely held and the gripper ceases movement. The experiment should include a diverse set of objects varying in shape, size, and weight, tested under controlled conditions to maintain consistency. Conducting multiple trials ensures reliable results. Grasp time can be recorded manually using a stopwatch to track the time from when the gripper starts moving until the grasp is complete. The average grasp time is calculated by summing the recorded times from all trials on each object and dividing the total

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

by the number of attempts. This per-object average grasp time is obtained on each of our 10 different objects and same is depicted in Fig. 18.

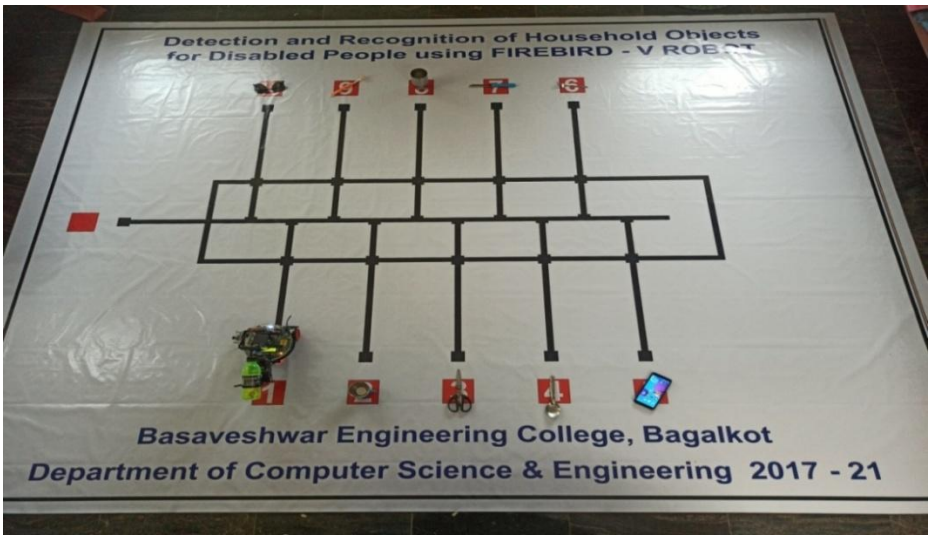

Figure 16: Robot Grasping an Object

3.3.4 Execution Time

For real-world applications, the system must operate in real-time. A lower execution time ensures faster response, making the robot efficient for real-time household applications. The execution time measures how fast each stage of the pipeline operates, including:

- **Detection Time:** Time taken by YOLO to detect objects.
- **Recognition Time:** Time taken by one of the proposed systems to classify the detected object.
- **Grasp Planning Time:** Time needed to compute the optimal grasp .
- **Total Inference Time:** The sum of all steps from image input to grasp execution.

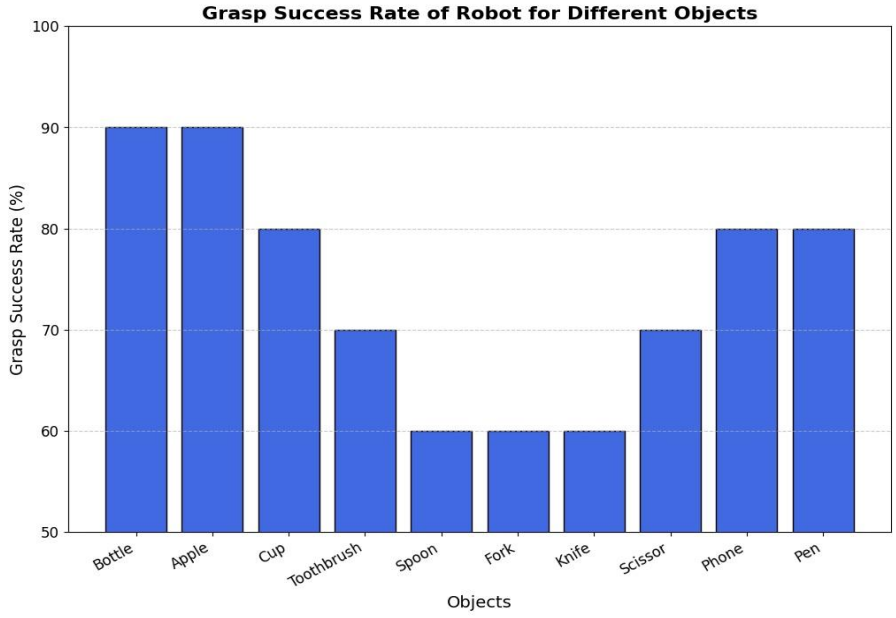

Figure 17: Object Wise Grasp Success Rate

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

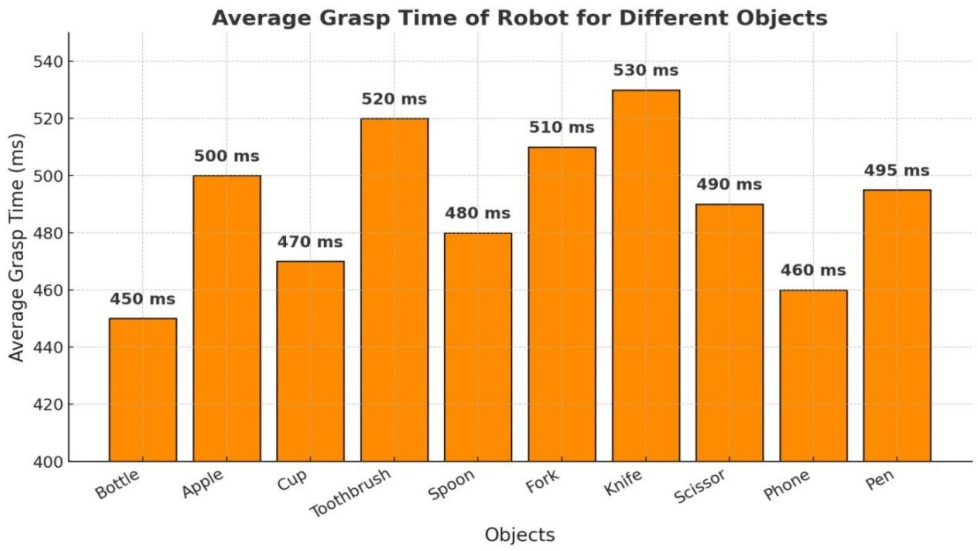

Figure 18: Object Wise Grasp Time

To obtain an average execution time, the results from multiple trials are summed and divided by the number of attempts. Optimizing this process can involve improving object detection and recognition models for faster recognition, fine-tuning gripper speed and force for efficient grasping. This per-object average grasp time is obtained on each of our 10 different objects and same is depicted in Fig. 19. Also Fig. 20 shows the task completion by Fire bird V robot.

In summary, this paper contributes to the ongoing discourse on the advancement of robotic perception and manipulation capabilities, with a focus on the Firebird V Robot as a platform for real-time object detection, recognition, and grasping. By evaluating the performance of these functionalities, we aim to facilitate further research and development in this domain, ultimately paving the way towards intelligent robotic systems that can seamlessly integrate into our daily lives.

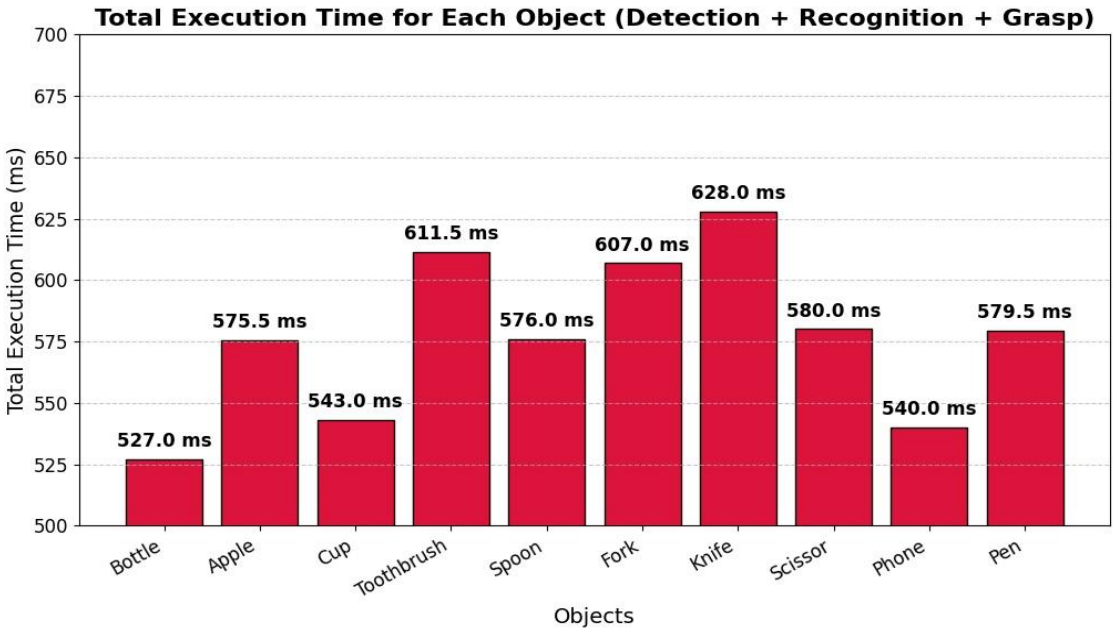

Figure 19: Total Execution Time

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

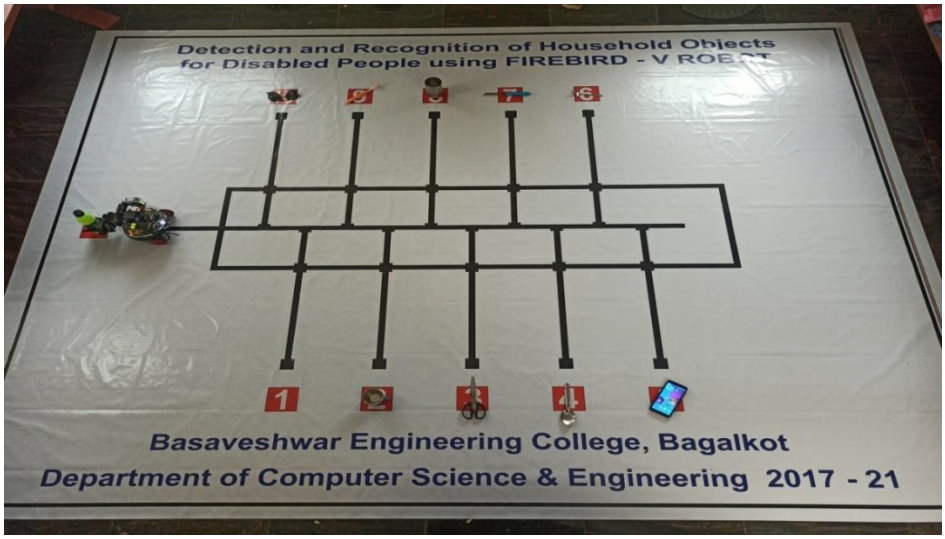

Figure 20: Robot completing its task

CONCLUSION AND FUTURE WORK

The real-time performance evaluation of household object detection, recognition, and grasping using the Fire Bird V robot demonstrates its effectiveness in autonomous manipulation tasks. YOLOv8n was employed for object detection, offering a balance between speed and accuracy, making it well-suited for real-time applications. The recognition performance was analyzed using four different methods: RGB-BPNN, PLC-SVM, RGB-CNN-SVM, and RGBD-CNN-EDL. Among these, RGBD-CNN-EDL exhibited the highest recognition accuracy due to its ability to utilize depth information and evidential deep learning, making it more robust against lighting variations and occlusions. RGB-CNN-SVM also performed well by leveraging deep feature extraction combined with SVM classification. In contrast, PLC-SVM and RGB-BPNN showed lower recognition accuracy, with PLC-SVM offering faster computation but lower robustness and RGB-BPNN being more prone to misclassifications due to its reliance on backpropagation-based learning.

The results indicate that the robot achieved an average detection time of 38.2ms, an average recognition time of 36ms, and an average grasping time of 490.5ms, with a grasp success rate of 74%. The grasping performance is influenced by object properties, with irregular shapes and reflective surfaces posing challenges. Optimizations such as refined servo motor control and adaptive grasping strategies will help to improve grasp success rates.

Although the Fire Bird V robot demonstrated promising real-time performance, several areas for improvement remain:

1. **Enhanced Grasping Strategies:** Implement adaptive grasping techniques using reinforcement learning to improve success rates for objects with varying shapes and textures.
2. **Real-Time Recognition Optimization:** Explore transformer-based vision models like Vision Transformers (ViTs) or Swin Transformers to further enhance object recognition accuracy and robustness.
3. **Multi-Sensor Fusion:** Integrate additional sensors (e.g., LiDAR, thermal cameras) for improved perception and scene understanding, enabling better object recognition under challenging conditions.
4. **Dynamic Environment Adaptation:** Develop AI-driven strategies for real-time obstacle avoidance and dynamic object manipulation, making the system more versatile in cluttered environments.
5. **Edge Computing for Real-Time Processing:** Implement edge-based inference using hardware accelerators (e.g., NVIDIA Jetson) to reduce latency and improve computational efficiency.
6. **Human-Robot Collaboration:** Extend the system to enable real-time human interaction, allowing users to provide corrective feedback during grasping tasks for improved adaptability.

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

By addressing these aspects, the Fire Bird V robot can be further enhanced to perform more complex and dynamic household tasks with greater accuracy, speed, and reliability.

REFRENCES

[1] Vaishnavi, K. & Reddy, G. & Reddy, T. & Iyengar, N. & Shaik, Subhani. (2023). Real-time Object Detection Using Deep Learning. Journal of Advances in Mathematics and Computer Science. 38. 24-32. 10.9734/jamcs/2023/v38i81787.

[2] Gede Putra Kusuma, Evan Kristia Wigati, Edward Chandra, A Review of Recent Advancements in Appearance-based Object Recognition, Procedia Computer Science, Volume 157, 2019, Pages 613-620, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2019.08.227.

[3] Hamidreza Kasaei, Mohammadreza Kasaei, MVGrasp: Real-time multi-view 3D object grasping in highly cluttered environments, Robotics and Autonomous Systems, Volume 160, 2023, 104313, ISSN 0921-8890, https://doi.org/10.1016/j.robot.2022.104313.

[4] Dzedzickis, A.; Suba˘ciut¯ e-Žemaitien ˙ e, J.; Šutinys, E.; ˙ Samukaite-Bubnien ˙ e, U.; Bu˘cinskas, ˙ V. Advanced Applications of Industrial Robotics: New Trends and Possibilities. Appl. Sci. 2022, 12, 135. https://doi.org/10.3390/ app12010135.

[5] Madhan, K., Shagirbasha, S., Kumar Mishra, T. and Iqbal, J. (2023), "Adoption of service robots: exploring the emerging trends through the lens of bibliometric analysis", International Hospitality Review, Vol. ahead-of-print No. ahead-of-print. https://doi.org/10.1108/IHR-12-2022-0058.

[6] Hernández, A.C.; Gómez, C.; Crespo, J.; Barber, R. Object Detection Applied to Indoor Environments for Mobile Robot Navigation. *Sensors* 2016, *16*, 1180. https://doi.org/10.3390/s16081180

[7] Megavath, Ravinder & Indra, Gaurav & Al-Rasheed, Amal & Alqahtani, Mohammed & Abbas, Mohamed & Almohiy, Hussain & Jambi, Layal & Soufiene, Ben. (2023). Indoor Objects Detection and Recognition for Mobility Assistance of Visually Impaired People with Smart Application. 10.21203/rs.3.rs-2814782/v1.

[8] M. Farag, A. N. A. Ghafar and M. H. ALSIBAI, "Real-Time Robotic Grasping and localization Using Deep Learning-Based Object Detection Technique," 2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), Selangor, Malaysia, 2019, pp. 139-144, doi: 10.1109/I2CACIS.2019.8825093.

[9] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[10] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Advances in Neural Information Processing Systems (NIPS).

[11] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. European Conference on Computer Vision (ECCV).

[12] Varma, S., Saini, M., & Nalbalwar, S. (2015). Firebird V: An Open Source Wireless Educational Robot. Proceedings of the 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC).

[13] Satpathy, S., Patra, S., & Pradhan, S. K. (2017). Real-time Object Detection and Tracking Using Firebird V Robot. International Journal of Control Theory and Applications.

[14] Behera, S., Rout, A., Mohanty, S. K., & Parhi, D. R. (2019). Development of Firebird V Robot for Educational and Research Purpose. 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS).

[15] Bisk, Y., Javdani, S., & Hager, G. D. (2016). Adapting Deep Visuomotor Representations with Weak Pairwise Constraints. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[16] Eppner, C., Grinblat, G. L., Furrer, F., & Siegwart, R. (2017). Real-time Probabilistic Grasp Detection with a Convolutional Neural Network. Proceedings of the IEEE International Conference on Robotics and Automation (ICRA).

[17] Zeng, A., Song, S., Welker, S., Lee, J., Rodriguez, A., Funkhouser, T., & Xiao, J. (2018). Learning Synergies between Pushing and Grasping with Self-supervised Deep Reinforcement Learning. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Research Article

[18] Basiri, M.; Pereira, J.; Bettencourt, R.; Piazza, E.; Fernandes, E.; Azevedo, C.; Lima, P. Functionalities, Benchmarking System and Performance Evaluation for a Domestic Service Robot: People Perception, People Following, and Pick and Placing. Appl. Sci. 2022, 12, 4819. https://doi.org/10.3390/ app12104819

[19] Zhong, J.; Ling, C.; Cangelosi, A.; Lotfi, A.; Liu, X. On the Gap between Domestic Robotic Applications and Computational Intelligence. Electronics 2021, 10, 793. https://doi.org/10.3390/ electronics10070793

[20] Behera, S., Rout, A., Mohanty, S. K., & Parhi, D. R. (2019). Development of Firebird V Robot for Educational and Research Purpose. 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS).

[21] The Fire Bird V ATMEGA2560 Hardware Manual is available on WordPress.com. The manual is © NEX Robotics Pvt. Ltd. and ERTS Lab, CSE, IIT Bombay, INDIA. https://roboram.wordpress.com/wp-content/uploads/2016/03/fire-bird-v-atmega2560-hardware-manual-2010-12-21.pdf

[22] The Fire Bird V ATMEGA2560 Software Manual is available on WordPress.com. The manual is © NEX Robotics Pvt. Ltd. and ERTS Lab, CSE, IIT Bombay, INDIA. https://roboram.wordpress.com/wp-content/uploads/2016/03/fire-bird-v-atmega2560-hardware-manual-2010-12-21.pdf

[23] Sohan, Mupparaju & Ram, Thotakura & Ch, Venkata. (2024). A Review on YOLOv8 and Its Advancements. 10.1007/978-981-99-7962-2_39.

[24] S. Gour and P. B. Patil, "A novel machine learning approach to recognize household objects," *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, Paralakhemundi, India, 2016, pp. 69-73, doi: 10.1109/SCOPES.2016.7955543.

[25] Gour, S., Patil, P.B., Malapur, B.S. (2021). Multi-class Support Vector Machine-Based Household Object Recognition System Using Features Supported by Point Cloud Library. In: Sharma, H., Saraswat, M., Kumar, S., Bansal, J.C. (eds) Intelligent Learning for Computer Vision. CIS 2020. Lecture Notes on Data Engineering and Communications Technologies, vol 61. Springer, Singapore. https://doi.org/10.1007/978-981-33-4582-9_8

[26] Gour, S., & Patil, P. B. (2020). An Exploration of Deep Learning in Recognizing Household Objects. *Grenze International Journal of Engineering & Technology (GIJET)*, *6*.