**Abstract - This project aims to create software enabling robots to effectively follow predefined patterns, which is crucial for enhancing efficiency across industries. The significance lies in optimizing repetitive tasks, from manufacturing to healthcare, by implementing precise pattern adherence by robots. The approach involves algorithm design and implementation to guide robots in executing tasks aligned with predetermined patterns, ensuring accuracy and consistency. The software developed successfully enables robots to seamlessly follow prescribed patterns, patterns, displaying adaptability and reliability in executing tasks as intended. In conclusion, this research underscores the importance of pattern-based software in robotic systems, paving the way for enhanced automation and efficiency in various sectors reliant on precise, repetitive actions.**

INTRODUCTION

Intelligent robotics has ushered in a new era of technological advancement, significantly impacting various industries. This report introduces a project titled *Robotic Automation: Follow the Stars*, undertaken at the University of Oklahoma, which aims to develop innovative software enabling robots to follow predefined patterns with high precision. Such capability is vital for enhancing operational efficiency in sectors heavily reliant on using pattern recognition. A real-world use case for such a project would be implementing a tour guide robot that follows set markers as a guide throughout the tour.

By studying and developing algorithms that enable robots to follow set patterns effectively, this project represents a step forward in integrating pattern-based behavior into robotic systems. The *Intelligent Color Image Recognition and Mobile Control System for Robotic Arm* study presents a sophisticated approach to integrating color recognition in pattern-based behavior into robotic systems. The study adopts a three-tier architecture of the Internet of Things (IoT), using Raspberry Pi and web page database, to control and monitor a color image recognition system, demonstrating effective supervision of manufacturing plants and precise color identification of different workpieces. Objectives underscore the integration of sensing, computing, control, and communication technologies in modern manufacturing, aligning with the objectives of Industry 4.0.

Both projects emphasize the increasing role of precise pattern and color recognition in robotics, underlining the shift towards more intelligent, adaptable automation systems. The study’s success in a manufacturing environment provides a valuable reference point for our work in enhancing robotic efficiency across various sectors, further aligning with the broader goals of Industry 4.0. *Follow the Stars* relies on the unique pattern of three-star-shaped objects, uniquely colored to stand out from the environment. The robot is coded to detect these and follow them around until. However, while the shapes and colors do not change, the order of the pattern does increase the difficulty in for the robot's detection.

However, the project has not been without challenges. Our team encountered several technical difficulties, particularly with the RGB camera and device connectivity, which proved to be substantial hurdles in achieving our objectives. These challenges included issues with the Astra Orbbec Camera not being recognized correctly, logging and module connection problems, and difficulties in color detection due to discrepancies between expected and actual color outputs. Despite these obstacles, our team was able to make meaningful progress, notably in the depth sensor's capability and the "Follow-the-leader" functionality.

The following sections will delve deeper into the specifics of our methodology, the technical hurdles we overcame, and the significant impact that our research holds for the future of robotic automation in various industries.

APPROACHES AND SOFTWARE

1. Overview

Our project aimed to create software enabling robots to follow specific patterns, leveraging advancements in pattern-based robotics to improve operational efficiency across various industries. The multi-faceted approach focused on algorithm design and practical implementation within the robotic framework. The inspiration and baseline for our approach were partially drawn from a which involved OpenCV and ROS color detection in C++. Our project translated these concepts into Python, adapting them to our specific requirements.

Central to our project is the use of TurtleBot's, versatile robotic platforms that provide a practical and accessible means for testing and demonstrating our software's capabilities. These robots, known for their adaptability and ease of use, are instrumental in navigating the intricacies of pattern-based behavior. Furthermore, our project extensively utilizes the Robot Operating System (ROS), a flexible framework for writing robot software. ROS is a collection of tools and libraries designed to simplify the task of creating complex and robust robot behavior across a diverse set of platforms. By leveraging ROS, we have been able to implement and test complex algorithms efficiently, allowing for rapid prototyping and iterative development.

1. Algorithmic Foundation and Code Development

Our color detection system was grounded in the principles of computer vision and image processing, utilizing the Python programming language. The cornerstone of our approach was the script color\_detection.py, which served as the mainstay for our color detection algorithm.

Initially, the algorithm involved defining specific HSV (Hue, Saturation, Value) color ranges for target colors like Neon Green, Cyan, and Magenta. These ranges were essential in setting the parameters within which the robotic system could recognize and respond to these colors in its operational environment.

A key function in our script, detect\_color, was designed to process the images captured by the robot's camera and ascertain the presence of the predefined colors within them. This function employed a technique where it created a mask for a given contour in the image, calculated the average color within that contour, and then matched this average to our predefined HSV ranges to identify the color.

Another function, find\_star\_contours, was developed to detect specific shapes or contours in the for testing, this function was pivotal in enabling the robot to discern and follow the designated patterns.

1. Addressing Technical Challenges

Our project confronted several technical challenges, particularly in integrating the color detection mechanism with the hardware components, notably the RGB camera and the Astra Orbbec Camera. These challenges were manifested in issues such as the camera only capturing depth information and not the RGB data, and difficulties in concurrent operation of the 3D sensors and the Astra Camera.

One of the primary issues we encountered was the system's misrecognition of the Astra Orbbec Camera as a Kinect device. This misidentification led to compatibility problems and hindered the camera's functionality within our system. Additionally, we faced difficulties in logging and output, which obstructed our monitoring and debugging processes. The OpenNI module, a crucial component for camera integration, posed challenges both in detection and connectivity, further complicating the integration process. In the end we ended up not using the ROS image processing option. One of the final hurdles had to do with the issues of simulators camera usage where our files were originally set up to use information from competing launch files. This meant the resources would block access to the camera and not allow it to be used or shared for other purposes.

RESULTS

The software testing phase for pattern-based robotic behavior unveiled promising outcomes, showing the software's ability to help robots follow predefined patterns accurately. However, the project's trajectory was significantly influenced by a series of technical impediments, related to the functionality and integration of the RGB camera system, notably with the Astra Orbbec. These challenges necessitated a strategic reevaluation and adaptation of our methods and objectives, leading to a set of outcomes that, while divergent from our initial vision, provided valuable insights into the complexities of robotic automation.

Despite the setbacks, our project saw the successful implementation of the 'Follow-the-leader' feature. This functionality allowed the robot to track and follow any moving object using its depth sensor, demonstrating the system's potential applicability in real-world scenarios. However, due to the challenges with device and camera connections, we had to simplify our color detection and coordination mechanism. This simplification was a strategic adaptation to maintain the functional integrity of the system but limited our ability to incorporate a timed switch between patterns, a feature we initially aimed to achieve. Alongside the follow the leader, the project built upon the framework from Project two with several obstacle avoidance and stopping measures.

DISCUSSION

In this section, we delve into a detailed analysis of our findings from the "Robotic Automation: Follow the Stars" project, focusing on the software developed for enabling robotic pattern following. Through this reflective exploration, we aim to elucidate the nuances of our research, the adaptations made in response to technical challenges, and the emergent insights that have shaped our understanding of robotic automation.

1. Insights from Algorithmic Refinement

One of the key insights we gained was the critical importance of color accuracy in the pattern recognition process. Our initial tests, which relied on pre-set color codes, revealed discrepancies in the robot's ability to recognize and follow the intended patterns. This realization led us to refine our color detection algorithm, specifically focusing on adjusting the color codes to match better the actual output of our printing hardware, which was markedly darker than our initial desktop-based simulations. This meant that we lost some of the distinctness of the original color choice, which was harder to find in a natural environment, but in return, achieved better accuracy.

1. Image of Shapes

A group of colorful stars

Description automatically generated

**Figure 1 - Follow-the-leader digital pattern type three: Cyan, Neon Green, Magenta**

A group of colorful stars

Description automatically generated

**Figure 2 - Follow-the-leader printer pattern type 3: Cyan, Neon Green, Magenta**

1. Insights from Pattern Recognition Performance

The software's ability to follow patterns, although challenged by hardware limitations and camera integration issues, revealed important insights into the behavior and capabilities of our robotic system. In static tests using image stills, the robots demonstrated a commendable level of precision in following the adjusted color patterns. These findings suggest that, under controlled conditions and with calibrated color settings, the software can perform with high accuracy and consistency.

1. Implications of Hardware Limitations

Our project's journey was significantly impacted by the hardware limitations, particularly those related to the RGB camera functionality. These challenges underscored the importance of hardware-software constructive collaboration in robotics. The misidentification of the Astra Orbbec Camera, and the issues with simultaneous operation of 3D sensors and the camera were all pivotal factors that influenced our approach and results. These experiences highlighted the necessity for robust and compatible hardware systems to fully realize the potential of sophisticated software solutions in robotics.

1. Comparision to Literature

The *Leader-Follower with TurtleBot3* project at Florida Atlantic University involves creating a leader-follower tracking system using the TurtleBot3 platform. This technology is being integrated slowly into industries for path-following tasks. The TurtleBot3, equipped with LIDAR, follows a user by scanning legs and applying a filtering algorithm. The system could have broader applications, such as in defense or aid for individuals with disabilities. This project aligns with the broader aims of robotic autonomy and efficiency seen in studies like the *Intelligent Color Image Recognition and Mobile Control System for Robotic Arm* which focuses on real-time control and monitoring in smart manufacturing, using IoT and Raspberry Pi to track workpieces via color. Both projects demonstrate the potential of robotics in complex pattern recognition and following, enhancing automation and operational efficiency in various sectors.

FUTURE WORK

1. Embracing Advanced Machine Learning Techniques

In our quest to advance the capabilities of robotic automation, a pivotal area of future work involves the integration of sophisticated machine learning (ML) techniques. We foresee the implementation of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) as transformative steps in enhancing the robot's pattern recognition abilities. CNNs, with their proven efficacy in image processing tasks, will be instrumental in refining the robot's visual data interpretation, enabling it to recognize more complex patterns. Meanwhile, RNNs, adept at handling sequential data, will offer improved insights into the robot's movements and actions within dynamic environments. Additionally, the adoption of transfer learning, employing pre-trained models on extensive datasets, stands to expedite the training process, broadening the robot's pattern and color recognition capabilities with increased accuracy.

1. Enhanced Pattern Switching

A significant focus will also be directed towards developing advanced pattern-switching mechanisms. Our goal is to achieve real-time pattern recognition, allowing the robot to swiftly adapt to environmental changes, thereby enhancing its operational flexibility. This necessitates the implementation of algorithms that facilitate quick detection and switching between patterns, ensuring the robot's readiness to respond to dynamic scenarios. Furthermore, the integration of predictive analytics into our system is poised to revolutionize the robot's functionality. By anticipating pattern changes, the robot can prepare for seamless transitions, elevating its responsiveness and adaptability in various operational contexts.

1. Broader Application and Testing

Looking ahead, our research will extend into conducting comprehensive real-world experiments. These tests, aimed at diverse industrial environments, will provide invaluable insights into the software's adaptability and performance under varied conditions. We would plan to delve into the integration of advanced sensor technologies, such as sophisticated vision systems and tactile sensors. Such enhancements are expected to significantly improve the robot's interaction with its surroundings and its precision in executing patterns. Finally, embracing collaboration and open-source development will be a cornerstone of our future endeavors. By inviting contributions from a diverse pool of experts, we would aim to foster a collaborative environment that nurtures innovation and facilitates comprehensive improvements to our robotic system.

CONCLUSION

In conclusion, our future endeavors in robotic automation are set to be marked by a deep exploration of machine learning techniques, the development of dynamic pattern-switching capabilities, and broadened application testing. These efforts are not only aimed at overcoming the current challenges but also at unlocking new possibilities for innovative applications across various sectors reliant on precise and adaptable automation technology.

APPENDICES

The appendices of this report encompass comprehensive documentation vital to understanding the project's technical implementation. It includes the complete source code, ROS launch files along with detailed documentation. Additionally, an appendix is dedicated to delineating the individual contributions of each team member across the project phases, namely design, implementation, testing, and reporting. These appendices serve as valuable references to comprehend the project's technical intricacies and acknowledge each team member's role in its completion.

**Appendix A.**

**Launch File Documentation**

**A.1. Header and Arguments**

A computer screen with text on it

Description automatically generated

**Figure 3 – Code snippet from Launch File**

This code (See Figure 3) segment defines arguments for various TurtleBot configurations. It sets values for the base, battery, GUI display, stacks, 3D sensor, RGB camera, simulation mode, and serial port, allowing for flexibility and customization within the TurtleBot environment. It accommodates different hardware configurations by specifying default values and environmental variables for adaptability.

**A.2. TurtleBot Bring Up**

A computer screen shot of a program code

Description automatically generated

**Figure 4 - Code snippet from Launch File**

This code (See Figure 4) snippet initializes TurtleBot functionalities by incorporating launch files responsible for various aspects like robot setup, mobile base configuration, and optional netbook usage based on the presence of a battery. Arguments are passed to configure the base, stacks, 3D sensor, serial port, and battery, allowing flexible configurations for TurtleBot deployment.

**A.3. Launch the Astra Camera and TF2 Publisher**

A screen shot of a computer code

Description automatically generated

**Figure 5 - Code snippet from Launch File**

The provided code (See Figure 5) segment initiates the Astra camera through a namespace, incorporating an Astra camera launch file and specifying camera arguments. Additionally, it activates a TF2 publisher node from the 'robot\_state\_publisher' package, configuring its publishing frequency to 30Hz for robot state updates.

**A.4. Launch the Fake Laser and our Scripts**

A screen shot of a computer program

Description automatically generated

**Figure 6 - Code snippet from Launch File**

This code snippet (See Figure 6) configures nodes in a ROS launch file. It sets up a fake laser from a depth image, converting depth data to laser scan data. It includes nodes for robot control, color detection, and teleoperation, each executing specific Python scripts, outputting their respective screens for visualization and control in the robot system.

**Appendix B.**

**Teammate Contribution**

**B.1. Design**

Alex, Amina, and Averi collaborated extensively to develop the project's overarching design, pooling their expertise to create a comprehensive blueprint. Their equal contributions were evident from the project's inception, as they collectively outlined and established the initial framework within the first few days. Their collaborative efforts shaped the foundation, ensuring a cohesive and well-thought-out design strategy for the project's progression.

**B.2. Implementation**

**Follow the Leader Code**



**Figure 7 - Code snippet from Control Script**

This code(See Figure 7) shows the implementation to get the follow the leader code started. Simply press ‘f’ on the keyboard to initiate the follow the leader command.

A white screen with black text

Description automatically generated

**Figure 8 - Code snippet from Control Script**

This code (See Figure 8) is what executes after ‘f’ is pressed on the keyboard while the turtle bot is in autonomous mode. It will follow an object within 3-5 feet in front of its camera sensor, following the object within 3 feet for 30 seconds.

In the implementation phase, Alex focused on the development of the 'follow the leader' code, dedicating efforts to its entirety and conducting thorough testing to ensure its functionality. Simultaneously, Averi and Amina focused on refining the pattern detection aspect. Collaborating seamlessly, Averi and Amina navigated the intricacies of pattern recognition, collaborating for this crucial component of the project. Their collective efforts culminated in a adequate implementation, blending 'follow the leader' functionality with refined pattern detection capabilities.

**B.3. Testing**

During testing, Alex tested the 'follow the leader' functionality, confirming its seamless operation and validating its comprehensive functionality. Meanwhile, Averi and Amina focused their testing efforts on the pattern detection module. Their thorough testing regimen ensured the accuracy and reliability of the pattern recognition system. Together, their dedicated testing efforts verified the efficacy of both the 'follow the leader' feature, assured by Alex, and the precision of the pattern detection, rigorously examined by Averi and Amina.

**B.4. Reporting**

In reporting, Alex played a pivotal role in validating the project concept, ensuring its alignment with initial intentions. Amina maintained regular updates with the professor, ensuring alignment with project expectations. Averi took charge of communicating our progress to the professor, facilitating an extension to our deadline when necessary. Moreover, the entire team collaborated evenly on all deliverables, including the report, presentation, and poster, ensuring a cohesive and comprehensive submission representing our collective efforts and achievements.

BIBLOGRAPHY

[1] E. Elephant\_Robotics, O. Ollii, and system  Closed, “The implement the color recognition on mycobot,” ROS Discourse, https://discourse.ros.org/t/the-implement-the-color-recognition-on-mycobot/25755 (accessed Nov. 20, 2023).

[2] penghou620, “Penghou620/ros\_color\_detection: Opencv and Ros Color Detection,” GitHub, https://github.com/penghou620/ros\_color\_detection (accessed Nov. 20, 2023).

[3] “Turtlebot Robots: Follow the leader,” Florida Atlantic University, https://www.fau.edu/engineering/senior-design/projects/spring2022/turtlebot-robots-follow-the-leader/ (accessed Nov. 20, 2023).

[4] W. L. Yao and H. C. Chen, “Intelligent Color Image Recognition and mobile control system for Robotic arm,” *2021 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, 2021. doi:10.1109/ispacs51563.2021.9651124