ASEE AMERICAN SOCIETY FOR ENGINEERING EDUCATION

Paper ID #45760

Bridging Theory and Practice: Undergraduate Engagement in Computer Vision and Robotics

Mr. Ayush Vasu Gowda, Florida Atlantic University
Dr. Daniel Raviv, Florida Atlantic University

Dr. Raviv is a Professor of Computer & Electrical Engineering and Computer Science at Florida Atlantic University. In December 2009 he was named Assistant Provost for Innovation and Entrepreneurship.

With more than 30 years of combined experience in th

Juan David Yepes, Florida Atlantic University

Assistant Professor in Teaching

Bridging Theory and Practice: Undergraduate Engagement in Computer Vision and Robotics

Abstract

This paper presents the journey of a computer engineering undergraduate student venturing into the field of computer vision and robotics, with a focus on optical flow and its applications in detecting moving objects during translational camera motion. It highlights the technical challenges encountered, along with the strategies employed to overcome them. Beyond technical skill acquisition, the paper also emphasizes the development of interpersonal skills, such as teamwork and navigating adversity. Additionally, the paper offers educators a hands-on experiment that can be implemented in the classroom, enabling students to engage with the same concepts and gain practical experience in an accessible way. In this paper, we detail the progression of technical expertise, problem-solving abilities, and creative thinking fostered through exploration.

The student joined this project with minimal robotics knowledge and only a basic understanding of computer vision. He learned about theoretical mathematical algorithms developed prior to his involvement and was introduced to existing Python and Excel simulations. After learning the theory, the student assembled a HiWonder JetAuto Pro Jetson Nano robot, created an artificial 3D environment, developed a Python program using OpenCV, and implemented and verified the theories and simulations. He also recorded and processed relevant videos.

As part of a team consisting of a professor, an electrical engineering PhD student, and other undergraduates, the student participated in weekly meetings where they discussed various problem-solving ideas using both divergent and convergent thinking. These meetings facilitated the exchange of ideas, yielding multiple solutions. One such solution involved integrating a mobile phone with the HiWonder robot to capture and synchronize accelerometer and gyroscope data.

This paper traces the student's exploration, which extended beyond classroom learning into hands-on experimentation using state-of-the-art robotic systems. We suggest how to expand this success to many students interested in hands-on research. To make the concepts of optical flow accessible to students, we developed a hands-on classroom activity for classroom implementation. Students set up a smartphone or webcam camera to record videos of objects moving toward or away from it at various speeds. They then import these videos into their IDEs and apply optical flow algorithms like Lucas-Kanade or Farneback to visualize and analyze the motion patterns. By observing the optical flow vectors, students directly experience the effects of visual looming. This activity reinforces theoretical concepts, provides practical experience with image processing techniques, and fosters critical thinking as students troubleshoot and engage with advanced computer vision topics.

Introduction

Computer Science and Computer Engineering have emerged as defining forces of the 20th and 21st centuries, leaving their mark on human history with their levels of innovation and rapid progress [1]. These fields have not only revolutionized the way we interact with the world around us, but have also become attractive career prospects, supplying high-paying opportunities and intriguing projects [2].

Among the various branches of these disciplines, Computer Vision has recently garnered significant attention due to its ability to mimic human-like perception using computing technology. By employing algorithms and processing data, it enables machines to comprehend and engage with the visual world. This has broadened the use of computers in fields that are typically reliant on human visual and processing skills such as transportation, manufacturing, and healthcare.

Recognizing the immense potential for students that go into this field, this paper explores the educational trajectory of a Computer Engineering and Science student, in Computer Vision. It aims to highlight the potential for students in Computer Science and Engineering, focusing on the process of the student's progressive mastery of this field. Initially unfamiliar with the field, the student progressed to develop a low-complexity system for a robot to detect external motion unrelated to the robot's own motion while in transit, and publish a conference paper on the methodology.

Moreover, this paper seeks to provide educators with insights to more effectively guide their students to success in Computer Science and Engineering, such as addressing the challenge of bridging the gap between aspirational goals, practical limitations of technology, and students' knowledge base. This mismatch is often one of the reasons for a "hump"- a phase where students often face discouragement and lose motivation. However, overcoming this "hump" is crucial for students to truly understand the concepts they are working with and learn how to deal with similar situations in the future. This paper aims to give a detailed view into the learning process, triumphs, and pitfalls of an undergraduate student to allow educators to more effectively help their students.

1. Educational Goal

The goal of this project was to determine how an undergraduate student would approach and work on a high-level topic with minimal prior experience. Before joining the project, the student had some experience with Python equivalent to an introductory class, but was unfamiliar with visual looming. For this project, he learned visual looming as described in the paper by Raviv and Joarder (2000) [3].

The aim was to observe the student's learning journey as the undergraduate student would grasp the principles of visual learning, develop a Python implementation of the principles, be given a problem along with a theoretical solution, craft a script demonstrating this solution, and finally integrate this script into a robot to prove its effectiveness. The decision to provide the undergraduate student with the theoretical solution to the problem was intentional and based on a shift in focus from previous studies. While earlier studies, such as Macri et al. (2023) [4] focused on helping students develop solutions to problems and bringing those solutions to reality, this paper aims to find and teach strategies to deal with the challenges and obstacles students will face in every other aspect of project development. This will show how various strategies can be implemented throughout development to proactively address challenges. This preemptive approach helps in maintaining student motivation and ensures success, by providing guidance and support tailored to the needs at each phase.

Overall there are four main educational goals in this paper:

Understanding Visual Looming - This aspect focuses on educating the student about the fundamental principles of visual looming. It involves comprehending and quantifying how objects appear to grow larger as they approach closer, and how to utilize this phenomenon.

Proficiency in Technical Tools - The student will learn how to utilize industry-standard technologies and tools to assist them during project development.

- Programming with Python
- Employing libraries like OpenCV and NumPy
- Utilizing Git and GitHub
- Operating Linux

Communication and Information Dissemination - Given the student will initially be inexperienced with the project, and in a new environment with new mentors, this element is vital. For example, weekly meetings with mentors, while useful for the student to clear up doubts, will not provide the student with much value if the student is unable to articulate where their problem lies.

Learning how to Learn - The student will improve their learning and working methods week over week, allowing them to improve their ability to grasp and build upon information. This will help them become more productive and efficient as a researcher, overcoming any “humps” they come across and also develop strategies to help them overcome “humps" throughout their life.

In summary, the educational goals of this project were designed to go beyond teaching technical skills, and focus on enhancing the student's overall proficiency. The aim was to prepare them with a diverse skill set, fostering adaptability and readiness for professional environments beyond the classroom [5].

2. Visual Looming

A. Introduction to Team and Initial Learning Phase

Upon joining the team, the student was introduced to an environment free from immediate productivity pressures. This period was designated for learning, with minimal expectations for innovation or creation. This was made abundantly clear to the student to minimize any pressures on them. In their first meeting, the student received Dana H. Ballard's book "Computer Vision" [6] to help them acquaint themselves with the overarching topic at their own pace [7]. This approach emphasized understanding over immediate results, allowing the student to absorb the material, and seek clarification on confusing topics. The student was advised to not focus on memorizing the entire book. Instead, they were encouraged to explore the concepts at their own pace and they were encouraged to ask questions about topics that they found interesting without worrying about initial confusion on any topics. This approach aimed to establish a solid theoretical foundation, crucial for building upon it, and get the student interested in the subject-matter.

B. Transition to Hands-On Learning and Focus on Visual Looming

Then the student's learning path shifted to a practical approach under a PhD student's mentorship, focusing specifically on the field of visual looming. This concept describes the quirk of objects moving toward a camera to appear to move towards the edges, and increase in size while doing so. He was taught the principles behind it and educated on using 2D and visual looming to calculate whether an object is moving towards or away from a point. Here the student was given their first task, implementing the Farneback model in a python script, a technique for visual looming tracking [8]. This task required them to record videos to test and experiment with the script, offering a practical approach to understanding and applying the concept.

C. Mentor-Mentee Communication Dynamics

For any new relationship, both parties understanding how to effectively communicate with one another is vital as it facilitates smoother interactions. Initially, the student was hesitant to ask questions, however this was addressed by explaining the objective, and how we wanted to remove as many barriers towards that as possible. This transparency cultivated a trusting and open environment, enabling the student to gain confidence and become more proactive in asking questions.

Despite this progress, communication clarity became the area for improvement. The student had difficulty clearly articulating questions in a way the mentors could understand, leading to confusion and misunderstandings. To tackle this, mentors emphasized the importance of enhancing communication skills. They encouraged the student to prepare questions in advance, practice adding context, and be explicit in their communication. This strategy led to more effective mentoring sessions, enabling a focus on specific areas of confusion and reducing meeting fatigue. As a result, meetings became more efficient, with energy primarily directed towards learning and explaining material. This not only streamlined the sessions, but also helped the team to engage more effectively in problem-solving, ensuring a focus on areas needing the most attention and avoiding unnecessary repetition.

D. Visual Looming Theory

Visual looming refers to the phenomenon where objects appear to increase in size as they approach closer to the observer.

To effectively analyze visual looming within a three-dimensional space, it is easier to employ a coordinate system that mirrors the functionality and perspective of a camera. For this polar coordinates, consisting of theta (θ), phi (φ), and radius (r), were chosen.

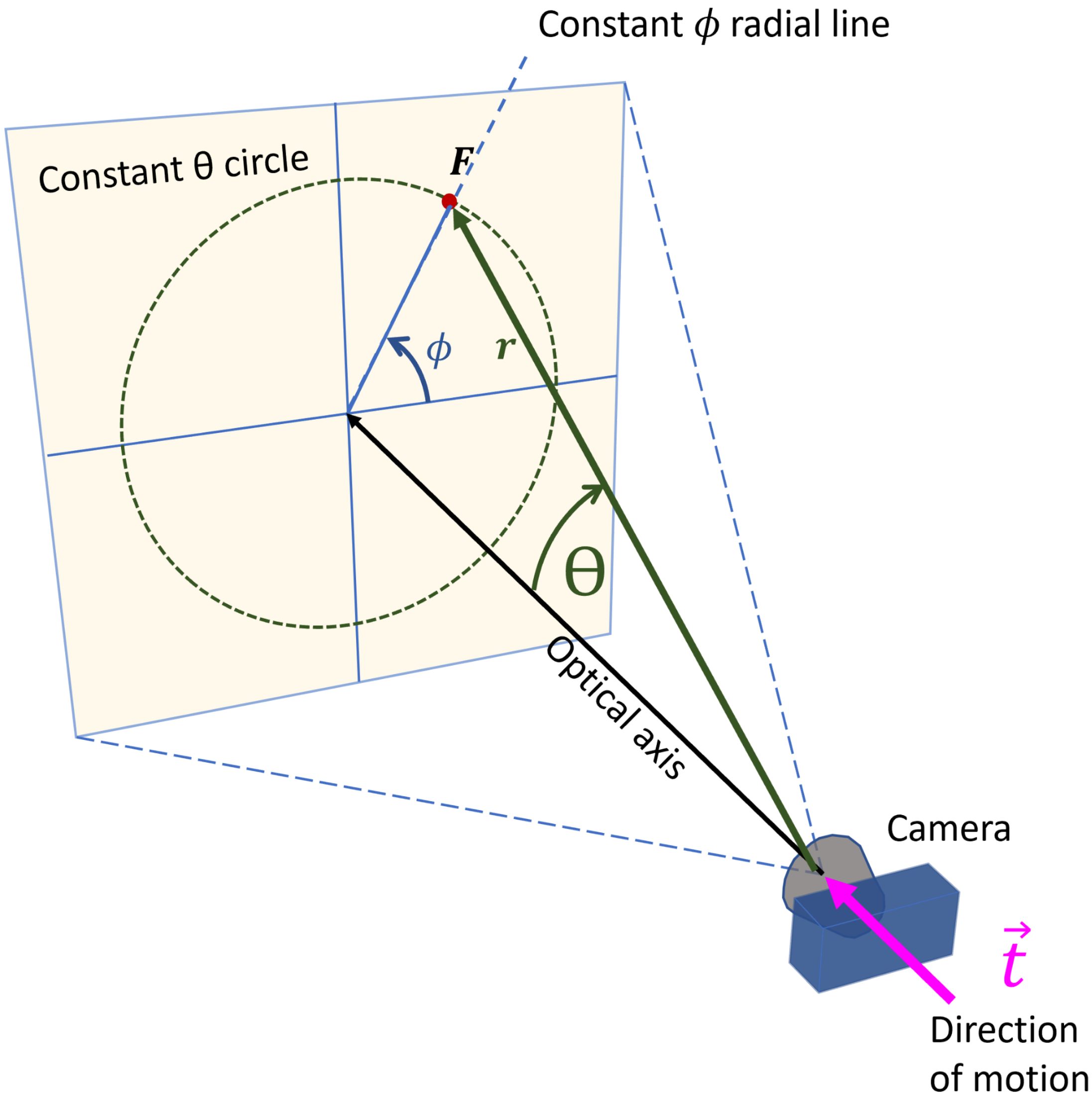

Figure 1 [9]: Diagram showing coordinate systems

1. Theta (θ): Represents the horizontal angle, theta is analogous to compass directions in a two-dimensional plane. It describes the left-to-right orientation of the camera's field of view.

2. Phi (φ): Represents the vertical angle, is perpendicular to theta. It signifies the up-and-down tilt of the camera, thereby accounting for the vertical aspect.

3. Radius (r): Denotes the distance from the camera to the point in space where the theta and phi angles intersect. Essentially measuring how far an object is from the observer or the camera lens.

Optical flow, another concept, allows us to measure visual looming, refers to the flow/movement of objects from one frame to another [10]. As it is inherently camera-based, the horizontal (x-axis) and vertical (y-axis) movement observed in pixels can be translated into corresponding theta and phi values. This translation allows for a more comprehensive and quantifiable understanding of object movement within the camera's field of view, particularly in analyzing the visual looming effect.

In scenarios involving linear or rectilinear motion, such as a vehicle moving directly towards the camera on a street, the relationship between theta and phi becomes increasingly direct. The degree of this change is directly proportional to the vehicle's velocity and its distance from the camera, embodying the concept of visual looming. By analyzing changes in theta and phi, one can infer the direction, speed, and thus the trajectory of an object's movement.

In summary, the use of polar coordinates with optical flow provides a low computation framework for quantifying visual looming and movement with a single 2D camera.

4. Problem and Application

A. Problem

After learning the basics of visual looming and creating a sample project, the student was presented with a new challenge. The task was to apply their understanding of visual looming and optical flow to develop a practical method for determining the movement of objects relative to an observer moving in straight, rectilinear motion. This involved using the ratio of two angles, phi (φ) and theta (θ), to identify the dynamics of object movement.

B. Application Theory

The proposed application uses the Polar Coordinate system, theta (θ), phi (φ), and radius (r). This problem focuses on an observer only moving in a straight line, and the principle of visual looming, where objects appear larger as they approach the observer, causing both θ and φ to increase proportionally. By analyzing the ratio of θ to φ, the solution can differentiate between objects that follow, and those that deviate from expected looming behavior—where the growth in width and height should follow a specific pattern across the observer's field of view and between frames. This method allows for the clear identification of independently moving objects within the observer's environment. This is because the objects' size does not change at the same rate as the background, and the specific pattern that all stationary objects follow, making them stand out. This approach stands out because in previous models, both the object and background would appear to move and generate optical flow. This would make it a challenge to find one moving object, let alone multiple moving objects [11].

5. The "Hump"

A. Climbing The "Hump"

The student began by programming a script that could perform the proposed solution, but soon encountered their first significant obstacle, and got stuck on this “hump”. This challenge arose from a lack of Python skills and specific knowledge about the Farneback model. The student had a hard time collecting the phi (φ) and theta (θ) values, and the student's lack of experience led to them feeling overwhelmed and unable to make progress.

Addressing procrastination by understanding and addressing its underlying reasons significantly boosts productivity and enthusiasm. This skill is crucial, not only for educators in guiding students, but also for students to develop independently. Self-awareness and proactive habits in overcoming procrastination can greatly benefit students throughout their lives, allowing them to work through any problem that they come across.

In this situation, the student expressed feeling stuck, and the mentors drew up a plan with the student on how to get past this roadblock. The plan involved initially concentrating on improving the student's Python skills, particularly through projects that would teach the student how to effectively manipulate matrices using NumPy, given that the output of the Farneback model is in matrix form. Once the student gained familiarity with NumPy, the mentors had the student practice extracting matrices from the Farneback model. From here, the mentors had the student combine his new knowledge on manipulation and extraction of matrices to get the student up and over the hump that he was stuck on.

Through this approach, the mentors successfully broke the student out of the procrastination cycle. The mentors helped the student break the project down into smaller, more manageable segments, and then link these segments together as the student grew more confident in their abilities. By identifying patterns that led to procrastination, the student not only overcame the immediate challenge, but also learned to segment and tackle problems in other areas of the project independently. This strategy prevented the onset of procrastination, significantly boosting the student's enthusiasm, and contributing to the project's overall success.

B. Going Around the “Hump”

Having completed the programming aspect, the student now had to configure the robot to move in specific patterns while simultaneously capturing video. This aimed to demonstrate the applicability of the Visual Looming Transform in the real world. However, the student encountered difficulties saving recordings using the HiWonder JetAuto Pro robot's camera module. These issues were largely due to the student's inexperience with Linux distributions and the HiWonder camera modules. Although the student could use the segmented method to break down this challenge, the student instead adapted to the situation and screen recorded the camera module's display, saving that video instead.

This experience highlights the importance for students to have a clear end goal, the ability to adapt, and an understanding of how each step contributes to achieving the final objective. This segment of the project was broken down into these steps: capturing video input, running the simulation, and outputting results. When direct saving attempts failed to capture the video input, the student adapted and instead began screen recording the display, showing the camera's view instead. But this was only possible because the student understood why they were doing what they were doing. This creative solution not only saved hours that would've been spent sifting through code, but also kept the student motivated, maintaining momentum on the project.

6. Implementation

A. Hardware:

- *Tri-Fold Board:* Serves as the video background, adorned with random colored scribbles to increase visual complexity.
- *JetAutoPro Robot (HiWonder Shenzhen):* Primary test platform with an arm-mounted camera.
- *Robot Arm Camera:* Oriented parallel to the floor and perpendicular to the Tri-Fold board.
- *PVC Pipes and Two-Way Adaptors*: Construct a dedicated track, measuring ~5 ft by 11 in, for consistent robot movement.
- *USB Dell Keyboard & Logitech Bluetooth Mouse:* Provide external robot control.
- *iPhone with PhyPhox:* Captures ground-truth movement data.
- *Laptop:* Runs simulations and the Visual Looming Transform script.

B. Software:

- *JetAutoPro Built-In Modules:* Adapted for precise control, enabling keyboard-based movement inputs and camera feed display.
- *SimpleScreenRecorder:* Captures the robot's camera feed.
- *PhyPhox (iPhone App):* Collects accelerometer-based motion data.
- *Visual Looming Transform Script:* Implements the theory described above and in [9].
- *PyCharm IDE:* Python development environment.

C. Data Collection:

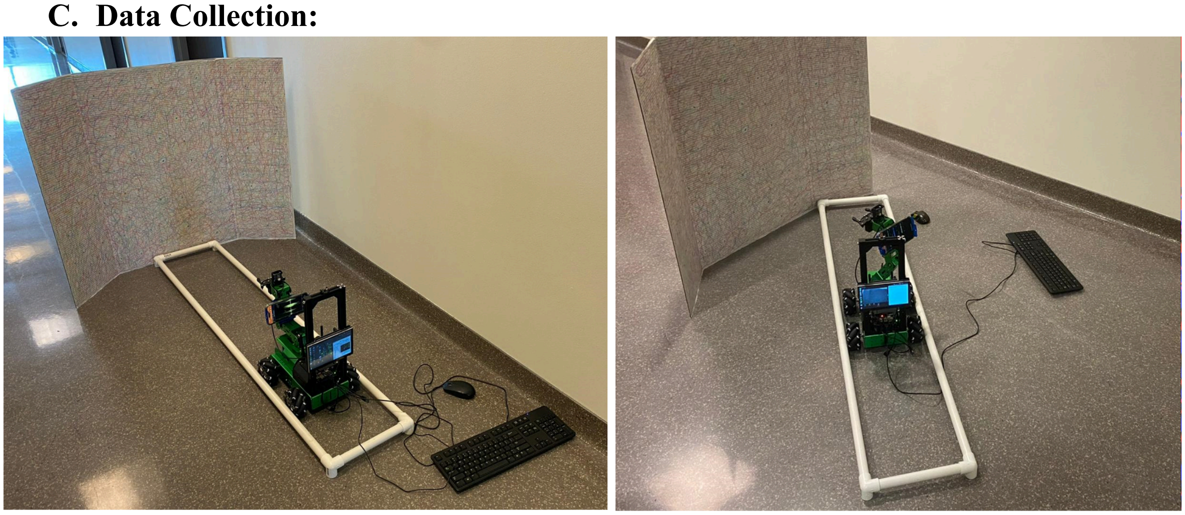

Figure 2&3: Robot movement on the track with the Tri-Fold board as background. The camera remains parallel to the ground and perpendicular to the board. On the left (Fig. 2), the robot moves perpendicular to the board. On the right (Fig. 3), the robot moves at a slant to the board.

Data Collection involved two movement patterns with the HiWonder Shenzhen JetAutoPro robot:

1. *Straight Movement:* The robot moved in a straight or nearly straight line with its arm-mounted camera oriented forward.
2. *Slanted Movement:* The robot advanced at a slant relative to the camera's orientation.

Both experiments were conducted on a PVC track to ensure consistent motion. The camera was aligned parallel to the ground and perpendicular to a static-patterned Tri-Fold board (Figures 2 & 3) [9]. Speed and video capture were controlled using adapted HiWonder modules. Calibration between video and accelerometer data was performed by physically lifting the robot along the Z-axis; however, Z-axis data was not used in the final analysis.

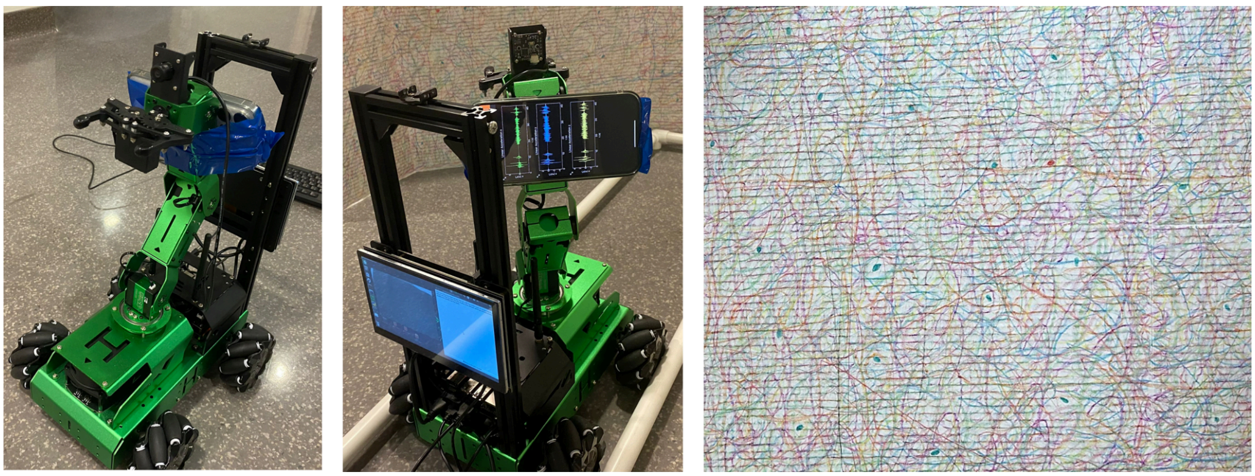

Figure 4 [9]: JetAutoPro robot with arm-mounted camera (left), back view and iPhone displaying PhyPhox (middle), and Tri-Fold board with simulated static (right).

D. Simulation Testing:

Figure 5 & 6: Stills from Unity simulations with a stationary background and a moving observer. Left (Fig. 5): Only background. Right (Fig. 6) Background with two motorists approaching.

Simulation testing was performed on Unity [12], and consisted of two types of tests in which the observer moved straight ahead. In one (Fig. 6) [9], there would be vehicles moving relative to the observer, and in the other (Fig. 5) [9], the environment was stationary without any vehicles. Simulation testing was not done by the student independently, but it is included to show the student's Visual Looming Transform script working as intended in a perfect simulated environment. This is because the research is ongoing and real-world results have yet to be published.

E. Data Analysis

Videos were cropped to uniform dimensions. Simulation videos were processed using the RAFT [13] method, while real-world tests utilized the Farneback method. The JetAutoPro's NVIDIA Jetson Nano struggled with RAFT's high computational demands, whereas Farneback's lower load allowed simultaneous program execution, robot movement, camera input, and result viewing.

Both RAFT and Farneback identify, visualize, and quantify optical flow. The student then used the Visual Looming Transform script to process the optical flow data by dividing phi and theta values for each pixel in every frame, extracting the background motion pattern. Deviations from this pattern indicated objects moving relative to the stationary background.

Computed values were compared with ground-truth accelerometer data from the iPhone and simulator to assess the accuracy of optical flow for motion detection and analysis. This comparative analysis is ongoing, and results are not included in this paper.

F. Experiment Results and Discussion

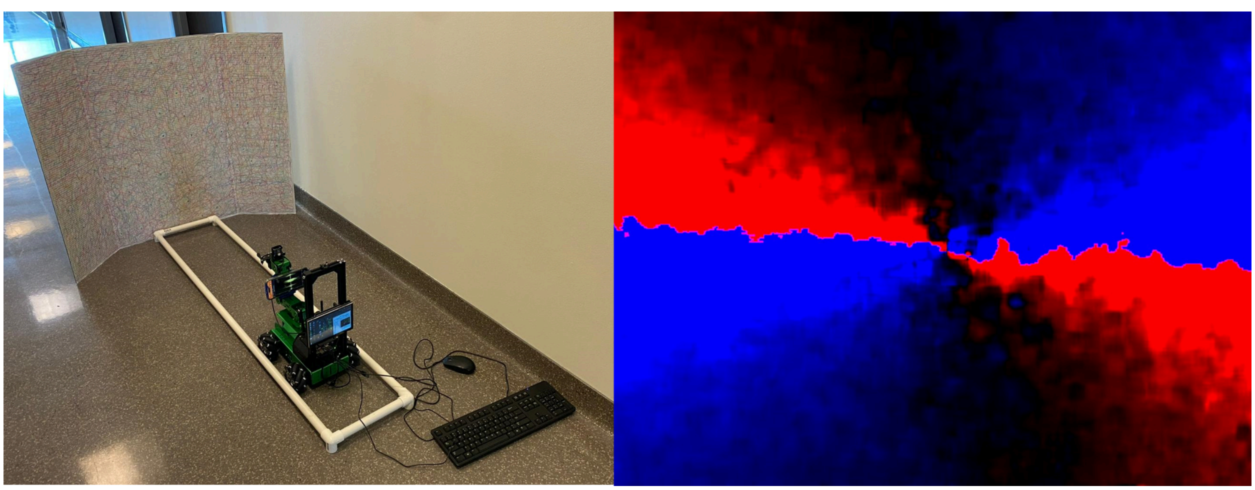

Figure 7: On the left, the robot is moving perpendicular to the Tri-Fold board. On the right, the Visual Looming transformation from this motion is shown.

Figure 8:On the left, the robot is moving slanted to the Tri-Fold board. On the right, the Visual Looming transformation from this motion is shown.

Real-world tests showed that the transformation results closely matched the theoretical pattern, demonstrating the method's effectiveness despite real-world noise. **Figure 7** displays the output, divided into four quadrants around the Focus of Expansion (FoE) [14]. The quirks of this image can be explained by division, which causes the middle vertical area to appear darker. This effect occurs because the calculation involves dividing a smaller number by a larger number, which approaches zero, leading to a darker and less intense color representation. In contrast, the central horizontal segment has a brighter appearance, due to dividing a larger number by a smaller one.

The pattern remains consistent during diagonal movements, as shown in Figure 8 [9]. Despite the robot's slanted trajectory, keeping the camera perpendicular to the Tri-Fold surface ensures the clear separation of quadrants and shows the slanted advance by shifting the focus of expansion. This consistency across different movements validates the Visual Looming Transform's reliability as a universal guide for detecting moving objects against a stationary background relative to a moving observer.

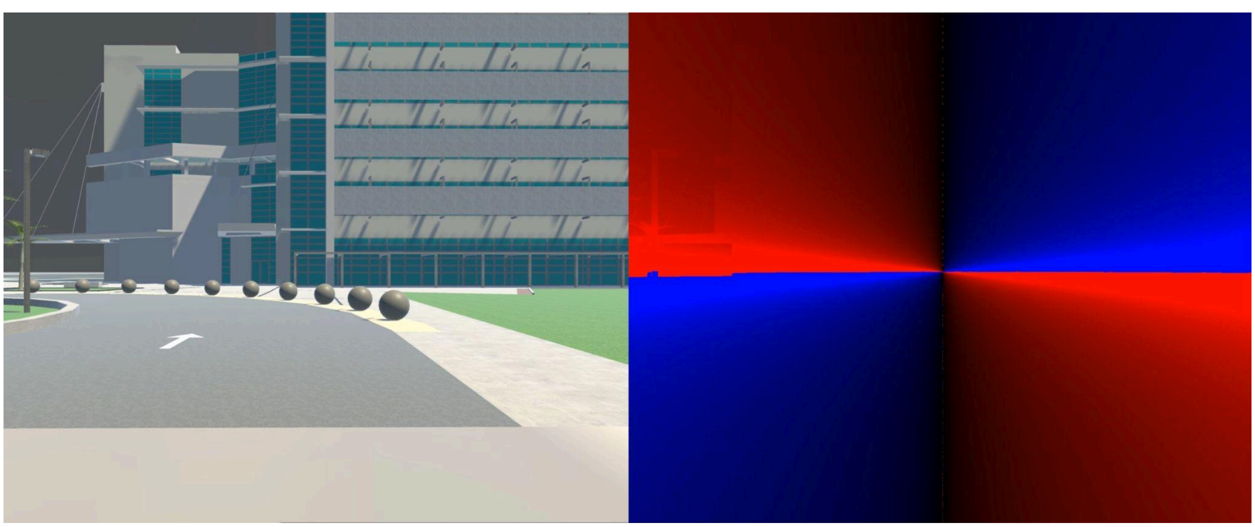

Figure 9: On the left, the observer is moving forward toward a stationary background. On the right, the Visual Looming transformation from this motion is shown.

Simulation testing further validates this concept. Figure 9 [9] shows a scene with only stationary background items, displaying the dark pillar seen in real-world tests. Conversely, Figure 10 [9] includes moving objects that are easily distinguishable by their contrasting colors against the background. The Visual Looming Transform is efficient and low-cost, requiring just a single 2D camera and minimal computing resources, offering strong potential for threat detection and obstacle avoidance.

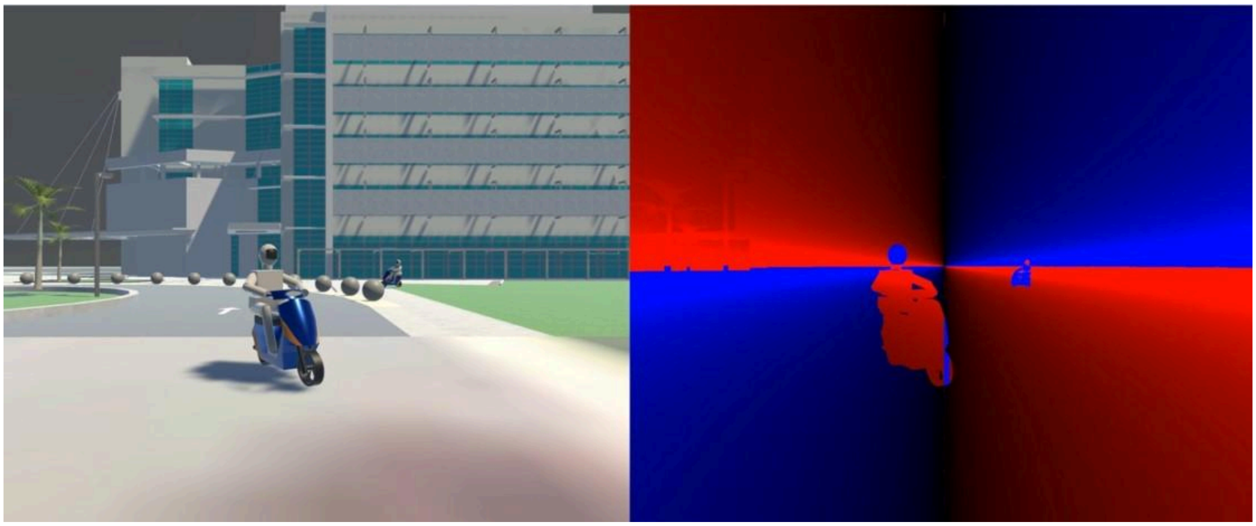

Figure 10: On the left, the observer is moving forward toward a stationary background with two motorists. On the right, the Visual Looming transformation from this motion is shown.

7. Student Educational Goal Outcomes

The overarching goal of this project was to observe how an undergraduate student with minimal prior experience would engage with and contribute to a high-level research topic, and to enhance that process to provide skills beneficial throughout the student's life and career. There were four specific sub-goals:

1. Understanding Visual Looming

A key objective was for the student to fully grasp the concept of visual looming and use it to detect moving objects. By the end of the project, the student had gained a solid understanding of the theory and successfully built on his understanding to create a functioning detection system.

2. Learning How to Learn

Another important goal was to help the student improve their ability to learn on their own. This included finding and understanding how to use resources like online tutorials, textbooks, and regular guidance from a professor and a graduate mentor. Through this, the student learned to break down complex uncertainties into smaller pieces and to go step by step in finding the answer to each of those pieces. He also learned how to optimize his questioning and knowledge seeking depending on the sources to get better answers.

3. **Proficiency in Technical Tools**

   Throughout the project, the student also became familiar with several key tools, including Python for programming and data handling, GitHub for version control and teamwork, and Linux for command-line operations and setting up development environments. These tools not only supported the student's work on this project but helped the student in passing interviews and securing jobs.

4. **Communication and Information Dissemination**

   Finally, the project placed a strong emphasis on clear and effective communication, both in writing and public speaking. The student wrote and co-authored a paper based on the project's findings, which was ultimately published, and presented these findings at several conferences. Through these experiences, the student learned to tailor technical content to different knowledge levels and to engage diverse audiences—ranging from peers with similar backgrounds to novices unfamiliar with the research area. This skill not only proved invaluable for communicating within the team and disseminating progress outside the project, but also enhanced the student's overall ability to convey complex ideas in an accessible and impactful way.

During and immediately after the project, the student received multiple awards for their research, presentation skills, and also technical abilities. These successes along with a newly enhanced skill to clearly explain his ideas led to several job offers before graduation. Notably, in one job interview, the student had to describe complex computer science topics in the form of processes he could implement to help the company reconcile erroneous chargebacks to nontechnical business owners. He used the communication and explanation skills learned during this project to make complicated ideas easy to understand—showing how the lessons learned over the course of a hands-on research project truly translated into real-world success.

Conclusion

In conclusion, this paper has explored the educational journey of an undergraduate student in the field of Computer Science and Engineering. The student started out as a beginner having never heard of Visual Looming or Optical Flow, and only had a basic understanding of Python. However, by the end, the student had progressed to a competent individual that had built a full system from theoretical concepts to a moving robot running a script developed by the student.

Moreover, the student developed the ability to navigate through difficulties, effectively communicate, apply theoretical knowledge to practical problems, and continuously improve their skills highlighting the importance of fostering an encouraging attitude in educational settings. The focus on "learning how to learn" and improving oneself week over week is a critical takeaway from this experience, and provides the student with a lifelong strategy to tackle problems, which is invaluable in the ever-evolving field of technology.

This paper also emphasizes the role of mentors and educators in guiding students through their educational journey. By providing support, encouragement, and real-world problems to solve, educators can significantly enhance the learning experience, helping students to bridge the gap between theoretical knowledge taught in classrooms and practical applications done in the workspace. This approach not only prepares students for successful careers in Computer Science and Engineering, but also instills in them the confidence and skills needed to tackle future challenges.

As the authors, we hope this highlights the need for educational programs that are not just focused on imparting technical knowledge, but are also committed to developing the whole individual, equipping students with the skills, mindset, and adaptability required to thrive in the real world. Through such programs, students can emerge as competent professionals, innovative thinkers, and problem solvers who can contribute more to the world while also succeeding in their future endeavors. To aid this we have created a simple hands-on experiment that can be implemented in the classroom: https://github.com/Hsuya01/Hands_On_CV_Exp.

References

1. Jesiek, B. K. (2013). The Origins and Early History of Computer Engineering in the United States. *IEEE Annals of the History of Computing*, 35(3), 6-18. https://doi.org/10.1109/MAHC.2013.2
2. Montoya, A. (2017). Computer Science for All: Opportunities Through a Diverse Teaching Workforce. *Harvard Journal of Hispanic Policy,* 29, 47-62. https://www.proquest.com/docview/1924502808?pq-origsite=gscholar&fromopenview=true&sourcetype=Scholarly%20Journals
3. Raviv, D., & Joarder, K. (2000). The Visual Looming Navigation Cue: A Unified Approach. *Computer Vision and Image Understanding*, 79(3), 331-363. https://doi.org/10.1006/cviu.2000.0862
4. Macri, V. A., Raviv, D., & Yepes, J. D. (2023, March). *From equations to actions: A system level design research experience of an undergraduate student* Paper presented at ASEE Southeast Section Conference, Arlington, Virginia. https://doi.org/10.18260/1-2--45012
5. Sarabipour, S., Hainer, S. J., Arslan, F. N., de Winde, C. M., Furlong, E., Bielczyk, N., et al. (2022). Building and sustaining mentor interactions as a mentee. *The FEBS Journal*, 289(6), 1374-1384. https://doi.org/10.1111/febs.15823
6. Ballard, D. H., & Brown, C. M. (1982). Computer Vision. Englewood Cliffs: Prentice Hall.
7. Yang, T. C., Chang Chen, M., & Chen, S. Y. (2018). The influences of self-regulated learning support and prior knowledge on improving learning performance. *Computers & Education*, 126, 37-52. https://doi.org/10.1016/j.compedu.2018.06.025
8. Farnebäck, G. (2003). Two-frame motion estimation based on polynomial expansion, In: Bigun, J., Gustavsson, T. (eds) Image Analysis. SCIA 2003. Lecture Notes in Computer Science, 2749, Springer, Berlin, Heidelberg, 363-370. https://doi.org/10.1007/3-540-45103-X_50
9. Raviv, D., Yepes, J. D., & Gowda, A. (2023). Detecting Moving Objects Using a Novel Optical-Flow-Based Range-Independent Invariant. In *2023 IEEE 20th International Conference on Smart Communities: Improving Quality of Life using AI, Robotics and IoT, HONET 2023* (pp. 137-139). (2023 IEEE 20th International Conference on Smart Communities: Improving Quality of Life using AI, Robotics and IoT, HONET 2023). https://doi.org/10.1109/HONET59747.2023.10374787
10. Horn, B. K., & Schunck, B. G. (1981). Determining optical flow. *Artificial Intelligence*, 17(1-3), 185-203. https://doi.org/10.1016/0004-3702(81)90024-2
11. Azim, A., & Aycard, O. (2012). Detection, classification and tracking of moving objects in a 3D environment. in *2012 IEEE Intelligent Vehicles Symposium*, 802-807. https://doi.org/10.1109/IVS.2012.6232303
12. Borkman, S., Crespi, A., Dhakad, S., Ganguly, S., Hogins, J., Jhang, Y. C., et al. (2021). Unity perception: Generate synthetic data for computer vision. *ArXiv*[Online]. https://doi.org/10.48550/arXiv.2107.04259
13. Teed, Z., & Deng, J. (2020). Raft: Recurrent all-pairs field transforms for optical flow. in *Comp. Vsn. – ECCV 2020 16th Eur. Conf.*, 402-419. https://doi.org/10.1007/978-3-030-58536-5_24
14. Jain, R. (1983). Direct computation of the focus of expansion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(1), 58-64. https://doi.org/10.1109/tpami.1983.4767345