Human Obstacle Detection and Tracking

|  |  |  |
| --- | --- | --- |
| Bharadwaj Chukkala  M.Eng. Robotics  University of Maryland  College Park, MD - 20740  [bchukkal@umd.edu](mailto:bchukkal@umd.edu) | Shelvin Pauly  M.Eng. Robotics  University of Maryland  College Park, MD - 20740  [spauly@umd.edu](mailto:spauly@umd.edu) | Venkata Sairam Polina  M.Eng. Robotics  University of Maryland  College Park, MD - 20740  [sairamp@umd.edu](mailto:sairamp@umd.edu) |

**Component description:**

This project involves the development of a Perception Module for Human Obstacle Detection and Tracking for a ACME Robotics product.

**Brief explanation of importance:**

Obstacle Detection has become an essential need in today’s technologically advanced world and the need for Automation has fueled this application. Assuming an industrial environment/warehouse setup where there is constant human-robot interaction on the ground. The industry contains automated guided vehicles for the transportation of mechanical/electrical components in an industry setup and products in a warehouse setup. There is an urgent need for a way to avoid imminent collisions between humans and AGVs. Although the pedestrians on the shopfloor/warehouse can be wary of the AGVs there has to be a way for the AGVs to detect human obstacles to avoid collision and collateral damage. We are going to develop a perception module that will be interfaced with the AGV to constantly detect and track moving humans. This ACME Robotics Product will use vision data alongside the sensor feedback, to avoid collision and choose a different path to traverse through the environment based on human detection and tracking from the continuous video input data stream, thus avoiding human collision by maintaining safe distance. The algorithm can be extended to detect other AGVs and Environment Variables so that the work floor is safe.

**Proposed Methodology**: The proposed module uses a higher level image representation, i.e., Histogram of Gradients(HOG). This higher level representation works as a feature extractor that works better than raw pixel comparison. We plan to use a Support Vector Machine(SVM) to detect humans in each frame. The detected humans are bounded by a box in the output frame of the detector. The separation is evaluated by the Intersection of Union(IoU) method. Finally, the centroid of the bounding boxes are used to track the movement of the humans. This data, is then transformed into the robot’s reference frame which can be used by the navigation system in the robot.

**Team Members, Roles and Responsibilities:**

* Venkata Sairam Polina (Driver)
* Bharadwaj Chukkala (Navigator)
* Shelvin Pauly (Design Keeper)

The roles are subject to rotation among the team members to encourage equal contribution and overall development experience for everyone. This switching will be done after a thorough discussion and analysis of skill and conceptual competency for the development phase.

**Deliverables**:

* A Perception Module designed and developed in C++ for a Monocular Camera, which will be used to detect Human Obstacles and Track them from a continuous video input.
* Software Version Control with Git and detailed commit history.
* UML Diagrams, Log Sheets with timestamps, Software Development Timeline.
* Continuous Integration and Code coverage using Travis CI and Coveralls
* Profiling and Memory Leak checking using Valgrind.
* Developer level Documentation using Doxygen.

**Software Development Strategies**: TDD, AIP, Pair Programming and Design Keeping

**Software Technologies**:

*Programming Language*: C++

*Development, Debugging and Testing Tools*:

* Makefile
* CMake
* cpplint
* cppcheck
* clangd
* Git
* Valgrind
* GTest
* Gmock
* VScode

*Operating System*: Ubuntu 20.04

*Documentation*: Doxygen

*Continuous Integration and Code Coverage*: Travis CI and Coveralls

*Libraries*: OpenCV 4.6.0

**Potential Risks and their mitigation**:

Improper labeling and training of a human detection model will cause issues in the functionality of the vision module and might lead to unprecedented circumstances. We are mitigating this issue by using transfer learning. ( backup: pre-trained model.

This project will be implemented as pair programming where there will be timely switching between driver and navigator roles, we mutually plan to share the design-keeping duty. Git version control will be used with consistent and clear commit history to track the updates. More than one branch will be created in the repository for the team to work and each pull request will be merged after thorough inspection by the significant other. Software development processes like TDD and AIP will be used.

**References**: TO BE WRITTEN