ENPM808X - Midterm Phase 0 (Proposal)

Kiran S Patil	Surya Chappidi	Vyshnav Achuthan
119398364	119398166	119304825

Overview:

ACME Robotics is planning to build an autonomous robot that requires a perception module that is capable of Human detection and tracking. Human detection and tracking are critical capabilities in the field of robotics, especially in scenarios where robots interact with humans or navigate in environments shared with humans.

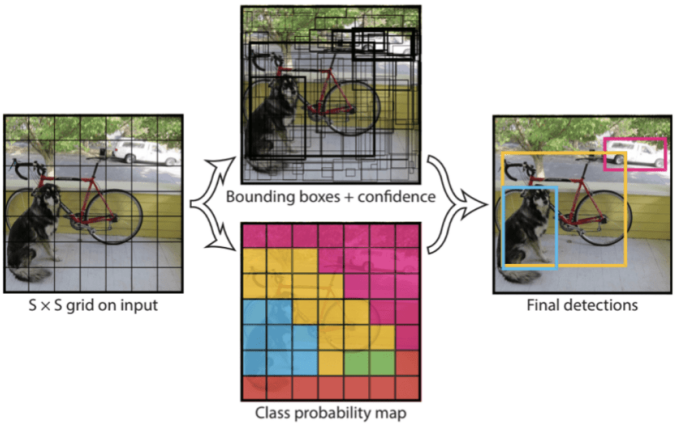
Assumptions:

The following are the assumptions considered:

1. Since we are using a monocular camera and have no information about the depth, we assume an average human height.
2. The lighting conditions are good.
3. Occlusion is absent.
4. Standard images are used for training and testing.

Design and Implementation:

The software will incorporate the YOLO (You Only Look Once) algorithm, which operates by partitioning the image into multiple grids. Each grid contains anchor boxes responsible for identifying and localizing various objects within that grid. Unlike conventional detectors, YOLO doesn't generate region proposals. Instead, it produces multiple bounding boxes that can later be refined through non-maximum suppression and IOU thresholding. The predictions encompass both class probabilities and bounding box information. This approach ensures swift and accurate detection since it involves only a single stage in the process. While there are advanced versions of this algorithm with enhanced accuracy and inference speed, we'll be utilizing the fundamental version for this project, which adheres to the same underlying principle.



Our intention is to utilize OpenCV Library's object tracking implementation to assess and validate the tracking quality. This will enable us to deliver a product that offers the highest standard of tracking, ensuring the excellence of our module.

We intend to employ the open-source OpenCV library to handle image input from the camera and perform necessary transformations and processing. Our selected programming language is C++. For building the software, we'll utilize the Cmake build system. To identify and address potential memory leaks, we'll employ Valgrind.

Development process:

We will adhere to the Agile Iterative Process (AIP) in conjunction with Test Driven Development (TDD) for our project. The team will operate in weekly iterations, maintaining a backlog chart to monitor progress. Our team comprises three members, each assigned specific roles as mentioned below. Development is carried out in a pair-programming approach. First, we focus on implementing requirements from the initial backlog, addressing bugs, and introducing new features. Any newly identified requirements are added to the backlog and subsequently addressed later. Git Version Control will be employed to meticulously track our progress. We will implement unit tests to verify code functionality, employing TDD principles to reduce errors. Additionally, we will prioritize sound coding practices and comprehensive documentation for effective project management.

Team:
Vyshnav Achuthan(Driver)
Kiran S Patil (navigator)
Surya Chappidi(Design Keeper)

Potential Risks and Mitigation:

- Low-quality input media - A high-quality camera and an environment with appropriate lighting conditions.

- False/Missed Detection - Enhancing the model's performance by training it on a more relevant dataset (comprising images/videos from the deployment area). This ensures that the model's parameters are finely tuned and the loss on the validation dataset is minimized.

Final Deliverables:

The module is designed to receive input media from a monocular camera mounted on the robot and furnish real-time coordinates of humans within the frame, relative to the robot's reference frame.

References:

[1] Software Engineering: The Current Practice, Vaclav Rajlich. CRC Press, 2011.
[2] You Only Look Once – a neural network for object detection