



# Responsible Robots and AI Via Moral Conditioning

**Omhier Khan<sup>1</sup>, Mark Alberta<sup>2</sup> and Dongbin Lee<sup>3\*</sup>**

<sup>1</sup>Department of Computer Science in the SOECS, University of the Pacific, USA

<sup>2</sup>Department of Communication in the College of the Pacific, University of the Pacific, USA

<sup>3</sup>Department of Computer Engineering in the SOECS, University of the Pacific, 3601 Pacific Ave, Stockton, CA, USA

**\*Corresponding author:** Dongbin Lee, Department of Computer Engineering in the SOECS, University of the Pacific, 3601 Pacific Ave, Stockton, CA, USA

**Received Date:** October 28, 2023

**Published Date:** November 17, 2023

## Abstract

This paper examines not only the main unresolved theoretical question, centered on the responsibility gap issue in autonomous robots, but also examines whether the robot can be responsible via moral conditioning and our proposed method bumper theory. Our scientific inquiry aims to discuss what is required to meet the goals and objectives of moral robotics, as well as promote dialogues and acquire quantitative results that hope to make robots responsible moral agents. Robots are morally conditioned with bumper theory by first detecting human-beings, dogs, or cars to avoid collision and reinforce prosocial morality. We adopted a scientific method such as deep machine learning and computer vision incorporated into a single board computer (Jetson Nano) to start detection and recognition processes of identifying objects including humans. This paper explores different models of computer vision and finds that Yolov4-tiny is the best use case on the constrained environment of the Jetson Nano to work towards building a system to solve this responsibility gap.

**Keywords:** robot; autonomous vehicle; self-driving; morality; moral conditioning; machine intelligence; machine learning; artificial intelligence; deep learning; Nvidia Jetson

## Introduction

Scientific thought has begun to seriously investigate the moral concerns that arise from robots, which are entities that perceivably will become a reality in our ever-technologically progressing society. Researchers tend to agree that one of the fundamental concerns derives from a “responsibility gap.” When responsibility meets ambiguity, the gap is formed, and this exists presumably already when the morally responsible agent is not always the robot and not always the one using the robot. Consequentially, the glaring concern centralizes itself around how responsibility gets placed when the robot does something potentially immoral. Commonly, the greatest concern is the killer robot, in which the robot kills a human being.

In this paper we seek to make sense of the ongoing conversation that has taken place around the topic and seek to come to a tangible proposal for how we can hope to resolve this responsibility gap between agents. What we suggest, and what we will entail, is the thorough moral conditioning of the robot. We seek to utilize moral conditioning to make the robot the morally responsible agent, and the one who bears the responsibility for its actions. We not only propose this as a theory, as has often been the case, but as a call to action for the application of robots. The scientific inquiry built around moral robots sorely needs a proposal for application on robots, if we are to resolve the concerns present around them.

To respond to the inquiry about responsible robotics, AI-enabled autonomous robots have been developed, especially vision-based robotic vehicles to avoid collisions (or accidents). In the field of Robotics and Artificial Intelligence (AI), ethics applied to machines are not well researched. One of the most popular forms of this ethics subject in robotics is Asimov's "Three Laws of Robotics". However, this does not cover many problems faced in autonomous robotic vehicles with artificial intelligence today such as the "Responsibility Gap". A responsibility gap arises when there is an absence of clear accountability. In this context accidents are caused by robots. To solve this responsibility gap problem, we decided to implement Johnathan Haidt's Moral Foundations Theory. To implement this social learning "rulebook" to our robot we need a highly reliable, and fast vision system to detect human-beings (including pedestrians) or vehicles to potentially protect them from accidents. This vision system must identify human bodies and human faces, if possible, to clearly discern collision that brings casualty from multiple ranges allowing for quick processing of context in the environment. Our proposal is testing human-being and faces in terms of speed or time delay and accuracy. Two distinct vision algorithms have been tested and we will choose the best combination of these two algorithms and run hybrid, or case by case in a variety of situations that sometimes require accuracy and others urgency. It consists of two; Multi-Task Cascade Neural Network (MTCNN) and You Only Look Once (YOLO).

For this task, MTCNN and YOLO are solutions to easily detect face and body. MTCNN would be used in face detection tasks allowing the robot to reach a speed of ~1 to 5 fps allowing for real time processing of environmental data and reaching an accuracy of 95% [1]. The YOLO family would be a great solution for high speed and body detection. Certain models of YOLO are able to achieve a mean Average Precision well above 50% and reach Frame Rates >100 [2,3]. We will explore both the YOLO Family and MTCNN Model to see if our AI robot can reach a confident base line for our ethics "rule book."

### Previous Literature on Robot Morality

Bigman et al [5] supposed that moral judgment hinges on perceived situational awareness, intentionality, and free will, plus human likeness. In their eyes, robots need to be aware that their actions are harmful and need to be held responsible. Robots need to be intentional agents and act independently. The human likeness component becomes important so that we can attribute responsibility to them. We find this paper important as one of the conversation starters of robot morality and urging considerations now rather than later for dealing with robot moral concerns. Robots are capable of making some kind of moral decision, they suggest, and rhetorically ponder if moral responsibility is going to exist within robots, do we really want to give robots moral rights?

Robillard [6] examined autonomous weapon systems (AWS), that possess features of being able to function independently of human operators, resist precise behavior prediction, and are capable of learning and adapting to environments. They mimic authentic decisions but are not genuine decision makers. Essentially, an AWS comes from the programmers' intentions, which makes it difficult

to suggest the robot is the morally responsible agent. Here is where the responsibility gap becomes apparent. If it is a programmer, then Robillard suggests it is hardly inspiring to create robots. If it is robots, then they are morally dangerous agents. Someone must have the responsibility, though, and Taylor [7] further exemplifies this gap when group agency is involved. For instance, if it is a government owned/programmed AWS, then is it the programmer(s), the robot, or the government who's to be the moral responsible agent? Someone has to be held responsible, face moral judgment and punishment, else Taylor suggests distrust in law or scapegoating might be further incentivized.

The responsibility gap even has gray areas on autonomous robots, as Jong [8] articulates. They cannot be fully controlled or predicted, making it difficult to place moral blame upon the autonomous robot. Too many factors are currently in place, such as who is observing autonomous car behavior during an immoral event, and some events are situational or simultaneous so how does responsibility become identifiable in those instances? Zhu et al [9] further addresses the moral blame concerns, that robots have the persuasive power to shape moral norms based on how they respond to human norm violations. Moral competency needs to be cultivated within the robot, creating essentially an artificial moral agency within them. For that to occur, robots would need to develop reflection skills and grow virtues.

There are those who have considered the ethical standard we would need to consider, in order to sufficiently qualify the robot as the moral agent. Danaher [10] proposed an ethical behaviorism theory, built around the performative equivalency of the robot. The "high" level standard in this theory would suggest that the robot is behaviorally sophisticated and is performatively equivalent to a competent adult human. While a low-level alternative is suggested, the above literature suggests that assigning objective moral blame to the robot would require the robot to unanimously be equivalent to a human. The theory also posits that we interpret the "empirical evidence concerning behavior" ([10]). Sullins [11] would suggest that behavior considerations fall under the autonomy, intentionality, and responsibility, some of which coincides with Bigman et al, but it's the responsibility portion that seems enigmatic. Sullins believes the robot must be fulfilling some social role, regardless of if it is conscientious of it or not. Behaviors and social roles, so far, seem to be the enigmas behind the responsibility gap.

### Machine/Deep Learning for AI-enabled Robots

The morality or ethics of Robotics & AI is a relatively new field with many experts debating on "concerns" over the topic. These patterns of concern for new technology can be traced back throughout human history, some concerns being quite silly, others pertinent. In the field of Robotics and Artificial Intelligence (AI), morality applied to machines is not well researched. One of the most popular forms of this morality subject in robotics is Asimov's "Three Laws of Robotics", However, the area is too broad to generalize the morality gap including, killer robots or weapons which is related to national interest and may be difficulty who is really responsible for the casualties. A responsibility gap arises when there is an absence of clear accountability. So, we would like to give the boundary for our ro-

bots to autonomous cars since such self-driving cars are often being issued or arguable matters of who are responsible to an accident because it is sometimes not clear whether the human-being including pedestrians or self-driving automakers made a mistake where the cars consist of thousands of parts with vendor companies. It is also ambiguous to charge the responsibility to such car user. This responsibility gap can bring turmoil if it is not clearly defined or make sure what we need to regulate as much as our society build the ownership of responsibilities. This still does not cover many problems faced in autonomous robotic vehicles with artificial intelligence today especially, how can we can put charge or penalty to the car like a human in terms of "responsibility" ownership where the accidents was caused by robots in this context?.

To accomplish this social learning "rulebook" to our robots under the AI-enabled autonomous vehicle systems, we need, a highly reliable, and fast vision system to detect human-being's face or whole body (pedestrians) or vehicles to potentially protect them from accidents. In computer vision there other broadly four problems to solve, but not limited to image classification which takes assign labeling to images, object detection where the object here is in the scope of human bodies, human faces or cars, segmentation or morphing, and object recognition including part of object recognition. This vision system must identify human bodies and human faces, if possible, to clearly discern collision that brings casualty from multiple ranges allowing for quick processing of context in the environment.

### Moral Conditioning in Autonomous Robotics and AI

We believe that scholars have aptly presented the problems at hand for ascribing moral agency to a robot, and also believe that a closer examination of moral theories alongside a practical application of them can help push the theories forward, in such a way that we can begin to utilize theory, not just converse it. Resolving the responsibility gap is of utter importance, considering we are welcoming robots of many kinds now and in the future. We propose that we can achieve autonomy, intentionality, and confidently place moral responsibility upon a robot, therefore making the robot the moral agent, by what we call moral conditioning. Robots, when made, are conditioned to perform tasks on a reliable level, just as much as a dog owner conditions the dog to reliably fetch a ball and return it to its owner. A task is placed upon the robot, and it is prescribed with a rulebook that must follow to a high percentage of accuracy.

The question becomes, considering the ambiguities present regarding the responsibility gap, what qualifies as an acceptable rule book for the robot? Considering the robot is likely to be a worldwide presence, therefore present in varying cultures, the moral rulebook must be based on universal moral decisions/behaviors. We believe that rulebook exists and intend on presenting those rules in the forthcoming sections of this paper. We then believe what's been missing is a proposition for how the rulebook can be programmed into the robot, how we can feasibly test empirically (like Danaher suggests) whether the robot is able to follow its rulebook at a reliable enough percentage that we can place responsibility on the robot and refer to it as a moral agent.

#### A. The Purpose (goals and objectives) of AI Robotics Morality Research:

- a. Condition a robot or robots to embody morality, to train the robot to make moral decisions in the likeness of human-beings - please refer to the below criteria based on reward and penalty,
- b. Research/experiment to develop sensing technology so a robot (or robots) can perceive and cognize human- being, cars, or region of interest,
- c. Apply AI approaches (Machine Learning, Deep Learning, Reinforcement Learning; shortly, ML, DL, and RL) capable of the robot growing in morality, seen through not harming humans.
- d. Optimal hardware & software selection to integrate engineering technology/scientific skills/sociality,

#### B. The objectives of this research: below are what we expect to accomplish and the objectives in skills to be learned, knowledge achieved, or performance expected:

- a. Develop machine intelligence such as deep vision sensing algorithms for object (human-being, car) detection or recognition,
- b. Enable AI approaches such as ML, DL, and RL skills to improve the level of autonomy,
- c. Strengthen vision-based deep convolutional neural network architecture to develop robot's learning,
- d. Improve the decision or execution in object recognition on an appropriate embedded hardware, such as Nvidia's Jetson NX, Nano series, or Beaglebone AI-64,
- e. Analyze the result of how the robot recognizes morality and update morality levels based on moral rulebook.

#### Moral Goodness and Badness

At the top-level moral concerns, the robot must be able to understand good and bad, or at least understand bad and not commit it. We are concerned about robots violating, not reinforcing, at this moment, as the above literature has us believe. We are concerned about moral violations in particular and committing the robot to being conditioned to avoid the violations with high accuracy. Nichols [12] suggested that norm systems help categorize moral violations, albeit on a human level they get complex. He also believes in the encoding of rules to help register these norm systems. Fortunately, we are not concerned about norm systems surrounding the morality of, say, marriage. What, then, can we be concerned about?

Our suggestion is Johnathan Haidt's Moral Foundations Theory (see Graham et al.) [13], as they have presented five universal moral foundations that humans are concerned with. In particular, and widely considered the most utilized, is the harm/care foundation. Though further down the line of programming moral rules, fairness/reciprocity, authority/subversion, loyalty/betrayal could also be considered, the above literature seems concerned with the harm aspect of robots. Similarly, to Isaac Asimov's three laws, here are three moral rulebooks based on Johnathan Haidt's Moral Foundation Theory.

- a. Care (Go)/Harm: This foundation is centered around empathy, compassion, and the concern for the well-being of others. It reflects our ability to care for vulnerable individuals and protect them from harm ("bumper theory").
- b. Loyalty/Betrayal: This foundation relates to group dynamics and social cohesion. It involves feelings of loyalty, duty, and belonging to a particular group, as well as aversion to betrayal or disloyalty.
- c. Authority/Subversion: This foundation focuses on respecting authority figures and maintaining social hierarchies. It involves adhering to traditions and norms that help maintain order and stability ("parent-child relation").
- d. Authority/Subversion: This foundation focuses on respecting authority figures and maintaining social hierarchies. It involves adhering to traditions and norms that help maintain order and stability.
- e. Sanctity/Degradation: This foundation is concerned with notions of purity, cleanliness, and sacredness. It involves avoiding actions or behaviors that could be considered polluting or degrading.
- f. Liberty/Oppression: This foundation revolves around the value of individual freedom and the aversion to oppressive actions or restrictions on personal autonomy.

For the scope of this paper, we will focus mainly on the Foundation of Care/Harm due to its straightforward implementation compared to the other 5 rulebooks. This is due to physical harm being immediate and easy to interpret. A robot with sensors that hits a person can process it much better than any other foundation such as Loyalty/Betrayal etc.

Giving a robot moral agency and responsibility is essentially making the robot resemble an artificial human agent. We have to then bring up the question of what human agency looks like and if we have to consider how much freedom we will deposit into the robot. To do so, we look at Roy Baumeister's [14] discussion of the philosophical disagreements built around human agency and free will. The interesting finding from Baumeister's discussion is how he examines what behaviors occur, rather than wrestle with the impossibility of defining objectively what free will ultimately is. We do want to avoid what Brems [15] emphasized as random behavior, in which the robot does not understand what it is doing and therefore makes a random choice. Instead, we are focusing on moral rules that are agreed upon by all cultures, that Baumeister sees as a foundation to guiding humans to making non-random choices.

Certainly, as he articulates, why humans follow moral rules is often religiously oriented. Certainly, the robot is not following religious orientation but is doing what humans have conditioned them to do. Realistically, Baumeister contends that "one's interest is to promote moral behavior so as to maintain the social system's ability to function smoothly and effectively" [14]. Baumeister revolves this contention around "free will as a moral agency that evolved for human moral culture" [14]. Understandably, then, we seek to condition robots to develop enough moral agency that they attain this degree of free will, that they can promote moral behavior

that maintains society's prosocial moral norm. Robots, then, avoid norm-violating behavior, which would be synonymously on the grounds of making random behavior, which goes against the above degree of free will. While we never seek to resolve the philosophical dimensions of free will, we do at least seek to condition the robot to abide by the moral free will described by Baumeister.

## Methodology

### A. Robot morality research: Bumper Theory and Experimental Learning with Machine Intelligence

The summary of our AI-enabled moral robotics is an integration of engineering/science with psychology and communication regarding robots, artificial intelligence, humans-machines interaction, and responsibilities because there are huge gaps in this integration system. The end goal is to build a kind a robotic system (autonomous system, for instance, self-driving vehicle; hereafter, we simply call "robot") that can "perceive, cognize, and recognize" the moral concept of harm/care through moral conditioning.

We will then test the robot's morality (level of morality) to demonstrate how the robot shall interact when used it in our society. To demonstrate the significance of the level of morality and clarify the responsibility gap in autonomous or self-driving vehicles including drones, we're going to build robots equipped with machine intelligence and computing algorithms based on many sensing systems similar parts of a brain of a human.

### B. Modality of Design and Evaluation

Mostly, perception & cognition-sensing abilities are crucial especially for robots to improve human-machine interaction and thus, to help resolve the responsibility gap and do research in this robotics morality. Our research is to focus on how the robots are to act and react in situations and experiments (sometimes environments) along with analysis of data, following DAQ obtained variety of sensors, which enables the robots to practice and resemble AI-enabled morality to perform harm/care behaviors. In time, AI approaches (machine learning, reinforcement learning, deep learning, deep reinforcement learning, etc.) are essential to make the autonomous robot morally inspired robotic systems or morally responsible.

- a. We first tested the robot to avoid hitting a human pedestrian or human being, without any feeling of guilt or responsibility, and increased warning of collision to stop the vehicle early. The robot, expecting to collide early on with the human, should then learn from its moral decision and future trials should see a decrease in collisions. The robot, then, should understand harm as the incorrect moral decision,
- b. Developed deep machine learning approaches to enhance the level of moral recognition with the autonomous vehicle,
- c. Increase the bumping distance (moral reasoning or morality) using machine intelligence approaches such as deep machine learning w/vision-based approach (CNN, YOLO, etc.) and deep reinforcement learning.
- d. To pursue research in the development of AI-enabled robots (ground vehicle that works in these goals). We Will develop re-



liable, cooperative human-machine, working to perform tasks collaboratively.

- e. The concept/degree of morality can be examined through reward and penalty and the coding for moral rulebook is adopting deep reinforcement learning,

A. If the robot (or vehicle)'s bumper hits a human or a car, the penalty will be added like car accident, depending on the seriousness of casualty. If more accidents occur, the penalties grow and cannot be run once the robot's penalty reaches at a certain penalty point for a while,

B. The reward will also be given like positive points if the robot has not gotten penalty for a while,

C. The moral awareness, or the level of morality, goes up as the reward goes up, the level will be down for the robot with higher penalty points, thus, the level of morality is proportional to the reward as it is inversely proportional to the penalty.

### Evaluation of Moral Rulebook with Morality Update with the Bumper theory

Norm learning, or norm conditioning for robots, is developed

to make the robot imitate social norms that are prosocial. The robot also has to understand norm-violating behavior, so as to avoid it. It then, and this may be the challenge, has to be able to judge these norms reliably in a variety of situations. Typically speaking, different norms would be assigned to different robots that fit a certain role. The decision-making process depends on how the judgments were conditioned onto the robot, that they make decisions in favor of social and moral norms and avoid norm-violating behaviors.

On a communication level, the robot should be conditioned to know when it has violated a norm, to accept responsibility for its action and take onto itself as the responsible moral agent. Conditioning robots to understand norms and violations, revolving around the foundation of harm/care, pertaining to a myriad of situations, helps give the robot a degree of moral agency and responsibility, to resolve this responsibility gap problem. We understand that the complexities of universal moral foundations are broad in itself. To focus the rulebook, we suggest an examination of moral and social cognition rules, as laid out by Voiklis and Malle [16]. The basis of this moral cognition approach is norm-learning, or acquiring norms that favor prosocial agency over antisocial agency. Based on above Voiklis and Malle, a modified step-by-step process of acquiring social and moral norms is suggested:

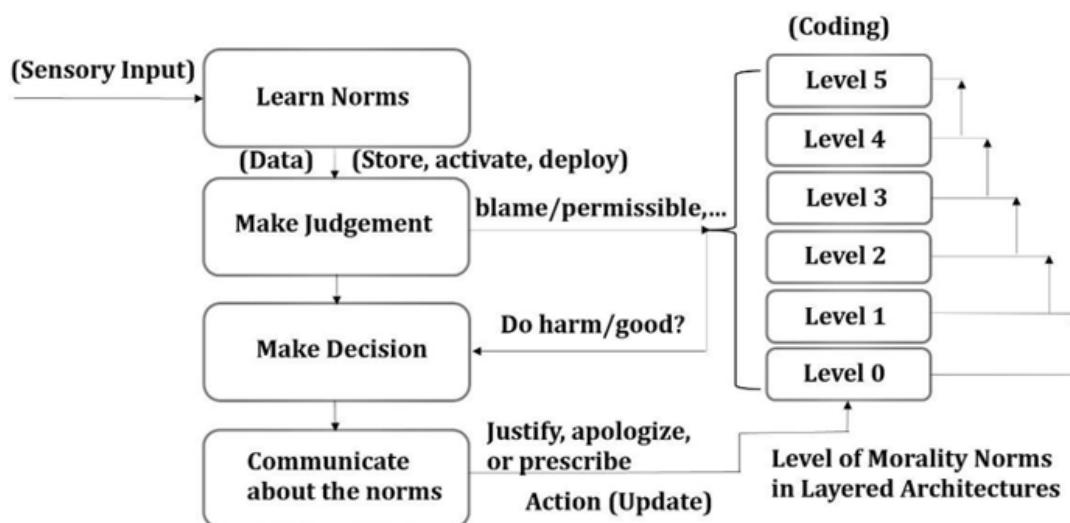


Figure 1: Moral Rulebook and Morality Update based on Bumper Theory.

The block diagram shown in Figure 1 conceptualizes how moral rule book shown in Sec. 2.1 and moral norms based on the bumper theory allows robots to engage and process together with the norms related to not harming people or cars. As the external sensory input is received the robot

- a. Learn, acquire, or activate norms, the robots,
- b. Make judgements to point finger, blame or permit about the norms based on the rulebook,
- c. Make decisions in the light of the level of moral norms in which each level is suggested by a bumper theory,

- d. Communicate about the norms to justify, apologize, or prescribe, and do in action to update the social norms step-by-step autonomously.

We suggest the level of morality from 0 to 5 based on normal adult human-being's morality, assumed level 5:

**Level 0** – nothing in moral sense, similar to a baby or kid. If the autonomous vehicle causes an accident, the bumper distance is negative because it hits the object (pedestrian, or a car) and the result of accident can be serious.

**Level 1** – a child level (or pre-school or elementary level): still

cause accidents driving with like a toy car, but the result can be serious, the bumper distance between the vehicle and human or another vehicle is said to be small negative value.

**Level 2** – middle schooler: can bring an accident or not often bring an accident even with a bicycle or a skateboard – the distance can be said zero.

**Level 3** – high schooler: able to drive a car, generally drive in structured environments, the accident would not often happen, some unconstructed situation may happen the accident.

**Level 4** – college level: pretty good but small number of positive (reward) value, the bumper distance is a bit away,

**Level 5** – general adult: considered as morally well-trained human-being, the reward is high, and the distance is far.

Note that our concern in this paper is to implement an autonomous vehicle system and machine learning using deep learning especially, to recognize human body recognition and further human face detection based on vision algorithms using MTCNN and YOLO to implement the level 5 which is our main concern, and the rest of levels is beyond the scope of this paper by replacing it with another paper conference paper. Currently, the proposed moral framework is a high-level objective. We first need to make our system detect the environment, mainly through computer vision models. We hope that the following experimental design provides steps towards resolving the responsibility gap issues with moral robots.

## Design of Experiments

The robotic system based on machine learning kits are consisting as follows:

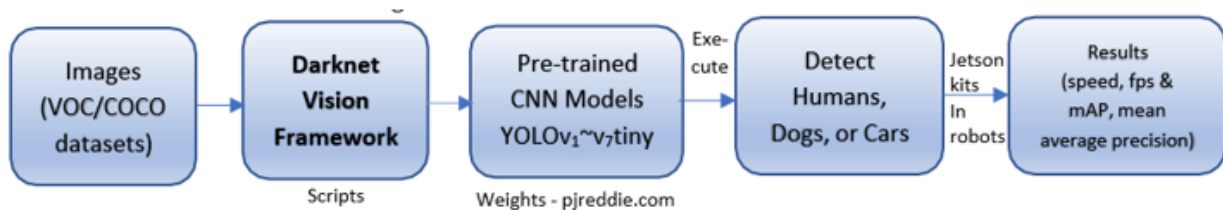
Ubuntu 20.0.4 as the Linux OS is running and the libraries are: TensorRT (modified models for Jetson nano), Nvidia CUDA & CUDNN libraries under Jeson Nano Kit, the scripts are Python 3, OpenCV vision libraries are adopted and, TensorFlow2 is running

for machine learning.

We are utilizing what can be termed Bumper theory. The theory suggests that the robot can make use of its bumpers as a data collection tool, in which it registers the collision as a morally wrong choice. Bumper theory is far more applicative to morally conditioning solo robots. Future research could improve approaches to Bumper theory, as well as explore other approaches.

- a. For machine intelligence, the embedded system in the robots use TensorFlow or PyTorch. Additionally, Google Coral Dev (TPU) board to accelerate the computational speed.
- b. We design the research first at the lab-based approach with small-scale autonomous mobile robots and humans, etc. and then, directly test them outside at the street.
- c. We utilize 5-Dim. Romi (Robotics and Machine Intelligence) Lab equipment and facilities for the experiment of autonomous robots to test the morally inspired robots and update the suggested criteria,
- d. The robot should at least compute the rulebook and able to do perception and cognition,
- e. Utilize at least an embedded computer (SBC) inside the robot such as Nvidia Jetson or Orion series, Raspberry Pi with Google Coral, or the like (SBC) to use for the main embedded computer,
- f. We will demonstrate the results with analytic tools mostly with quantitative methods.

The system consists of two distinct categories; detection under mainly Humans, Dogs, and Cars and face detection where the autonomous robot detects human-being first and identifies human face as the verification. A robot system for Human-being Detection and once detected, run the Face Recognition under Nvidia Jetson Lib./Ubuntu/TensorRT/OpenCV/Pip3 with a Logitech camera.



**Figure 2:** YOLO Computer Vision Approach to detect Objects.

## Simulation to simulations and Experiments

### A. The Framework for Human-being Detection.

In the results, the mean average precision (mAP) is used to evaluate the accuracy of human detection or recognition algorithms as a standard metric where the precision in mAP is responsible for the accuracy and of robustness of predictions. FPS (frame per second) is the speed to detect or recognize the target but the resolution in the simulation and experiments are higher (448x448 or 640x360) than, half of them or less as resolution-adjusted images.

### B. The Framework for Human Face Recognition

WIDER Face and FDDB are face detection datasets, of which images are selected from the publicly available datasets. WIDER FACE dataset is organized based on 61 event classes. For each event class, we randomly select data as training, validation and testing

sets, which consists of 393,703 labeled face bounding boxes in 32, 203 images. The plan of experiments is consisting of three experiments, and we demonstrated up to the second experiment like this: we have developed models for human face and body classification, detection, recognition approaches using Machine & Deep Learning for mobile robot's human or vehicle detection to test the vehicle to detect human-being first as pedestrian in the street. These approaches were tested with Nvidia's Jetson Nano, with vision using YOLO, with PyTorch and COCO dataset. These are targeted to develop experimental AI-enabled Robotic Agent's morality check to see if such intelligent robots or autonomous vehicles are morally responsible, not doing harmful actions on those groups and further to see how the robots can be conditioned to protect the objects while doing good while performing given mission interactively in a timely manner.

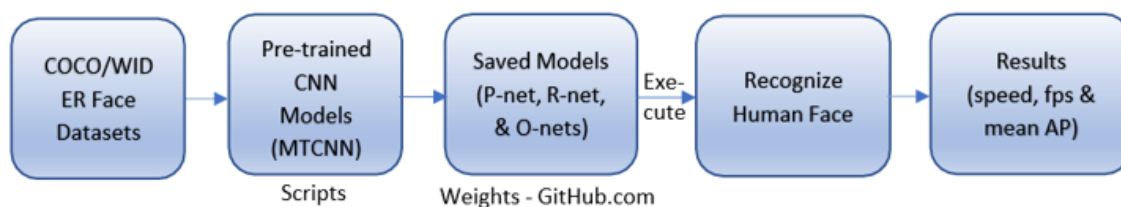


Figure 3: MTCNN Vision Approach to detect Objects.

## Machine Intelligence using Machine Learning with Deep CNN

Machine Learning has used YOLO for human recognition and multi-task cascaded convolutional neural network (MTCNN) for human-face recognition.

### Machine Learning and Computer Vision Constraints

For the computer vision system to be successful it needs to overcome constraints. These constraints are a low computing environment, real time fps, and relatively high accuracy. The low computing environment is due to our robot mainly being a small single board computer (Jetson Nano) allowing for limited processing. The robot also needs to have real time fps meaning it achieves close to 10-15 FPS allowing for semi smooth processing, and recognition. The last constraint is that accuracy has to be "relatively" high. This is due to the fact that the moral robot will carry out actions that can "harm" a human. Having it misidentify a person can be detrimental to our experimentation.

### Body Detection

Our next option is to implement Human body detection. Instead of focusing on the Face our end goal might work better using this type of detection, because of its general use towards vehicle movement. Human detection's main goal is to identify human beings in

a live video or static image. Latest Human body detection works similarly to Face detectors using deep learning algorithms.

Most body detection algorithms come in two types: single and two shot stage detectors. In two shot detectors there is a proposal stage and the refinement similar to MTCNN. This causes high accuracy however usually with a cost to speed famous detectors in this category include Faster-RCNN and FAST-CNN. Single Stage Detectors are on average faster and have decent accuracy. The detectors who go under this umbrella are You Only Look Once (YOLO), and SSD. YOLO seems promising with a wide family tree, and multiple use cases [2,3]. This allows for YOLO to be fit to almost any situation, especially our use case of using a constrained environment such as the Jetson Nano.

### YOLOv1 Introduction

YOLO (You Only Look Once) is a popular real-time object detection algorithm and framework for computer vision. It was introduced in a series of papers written by Joseph Redmon and Santosh Divvala [1,17,18]. YOLO is known for its speed and accuracy. Unlike traditional object detection methods that require multiple passes over an image and the use of complex sliding window techniques, YOLO frames object detection as a single regression problem. It divides an image into a grid and, for each grid cell, predicts bounding boxes and class probabilities simultaneously

[1]. Originally built for the Darknet Neural Network Framework, it is built in C and can be optimized for Nvidia GPUs which is perfect for our use case of using the Jetson Nano as our main Processing board for our experiments.

Three general Benefits of YOLO Network compared to other traditional networks as follows:

- Less Background Mistakes:** YOLO performs the convolution on the whole image rather than sections of it due to which it encodes contextual information about the classes and their appearances. It makes less mistakes in predicting background patches as objects as it views the entire image and reasons globally rather than locally,
- Highly Generalizable:** YOLO learns generalizable representations of objects due to which it can be applied to new domains and unexpected inputs without breaking,
- Speed:** YOLO is extremely fast compared to its predecessors as it uses a single convolution network to detect objects. The convolution is performed on the entire input image only once to yield the predictions.

There are now dozens of YOLO models implemented with different technologies, but for the focus of this paper we will cover the original YOLO, YOLOV2, YOLOV3, YOLOV3-Tiny YOLOv4, YOLOv4-Ti-

ny, and YOLOv7-Tiny.

## How YOLO works

Yolo first gets an image, and then Splits into multiple grids. This grid shape is  $S \times S$  and is usually determined by the designer. For the original paper it was decided to be  $7 \times 7$  by the author [1]. The smaller the grid the more details it can catch without the cost of computational speed. Then in the model each "cell" is responsible for outputting the midpoint of an object. For example, a Person can be in multiple cells, but the midpoint with the highest probability is determined through the picture. Each cell also has its own  $x$  and  $y$  coordinates starting from 0 going to 1. Allowing each output and label to be relative to the cell and allowing us to scale these points up to the actual image. The Label for each cell would have  $[X, Y, W, H]$ .  $X$  &  $Y$  correspond with the position of the midpoint in the cell while  $W$  &  $H$  will correspond to the width and the height of the box.

Each Cell has a full label that is  $B * 5 + C$  tensor. The  $C$  objects are the 20 classes in the VOC dataset [1].  $P_c$  this is the probability of a class inside of the cell, and  $[X, Y, W, H]$  is the positioning covered previously. For the Original design there are two Boxes in each cell that try to accurately predict if an object is wide or tall. Then all non-object cells are nulled, and the model focuses on the object cells. It applies IoU & NMS to combine boxes to get the overall bounding box for the overall object. Figure 4

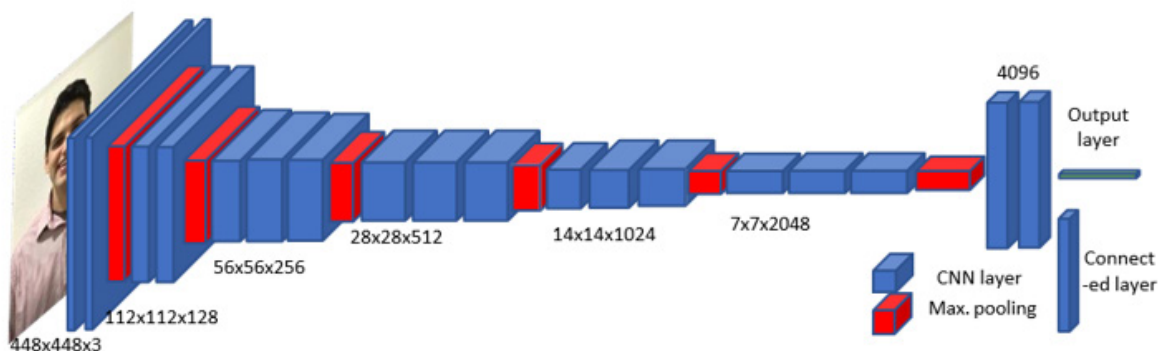


Figure 4: Convolutional Neural Network Architecture for Human Recognition and Face Detection.

## YOLO Architecture

This original YOLO design is inspired and based on the GoogLeNet architecture [1], in which YOLO has 26 Convolutional Layers, 4 max pool layers and 2 Fully connected Layers. In total there are 32 layers with the original architecture. All Layers use leaky rectified linear unit (ReLU) while the last layer uses linear activation function. The Initial convolutions extract features from the image, while the full connected layers at the end predict the output probabilities, and coordinates of each class. The Image is fed into the input layer of YOLO CNN network and cut into a cell grid ( $S \times S$ ). This structure allows the network to focus on the whole image. The final product of this architecture will be a  $14 \times 14$  grid. Each cell of the grid is based on 20 object classes from the VOC dataset, 2 Probability scores ( $C_1, C_2$ ), and 8 regression box points to create the bounding box.

## YOLO Training

For the first 20 convoluted layers they pre-trained with ImageNet 1000 class dataset however they only used 20 classes they were able to reach an accuracy of 88% they used  $224 \times 224$  feed layer [1,19]. This was used for classification. The main reason they used classification first, is that they found adding more layers to a classification network and changing it into detection makes it more accurate. When switching to detection they used  $448 \times 448$  images for finer details and added the last 6 layers 4 convoluted layers and 2 fully connected layers due to increase in performance. The network layers were trained using the VOC 2012 and 2007 dataset. With these images, modified to only include 20 classes and added center points to the image. This modified data will be elaborated shortly.





Figure 5: An Example of a VOC image.

X-Train  
(an Image)



Y-Train (Labeling  
for each cell)

Pc	1
Bx	50
By	70
<u>Bw</u>	60
<u>Bh</u>	70
C1	1
C2	0

Figure 6: (R) A Sample Trained Data using Traditional Neural Network for the Image.

### Traditional Neural Network

In a traditional neural network model, shown in Figure 6 an image is paired with a vector of numbers for training. The first element of this vector, Pc as the probability of object (class) inside the picture, is an object or not an object, the next 4 are the box dimensions [X, Y, W, H]. The last two numbers correspond with classes, in this case dog or person.

### How YOLO network is trained

In YOLO the training is done by taking the original VOC picture

and modifying it to fit the model. The first step to this is adding a specified grid, in our case a 7x7. Then after adding the grid, we manually put centers to the object for the YOLO model to train on see Figure 7. Then we pair all of the cells in the Neural Network with its corresponding values that were covered previously. In the example below three objects are labeled with their corresponding data labels.

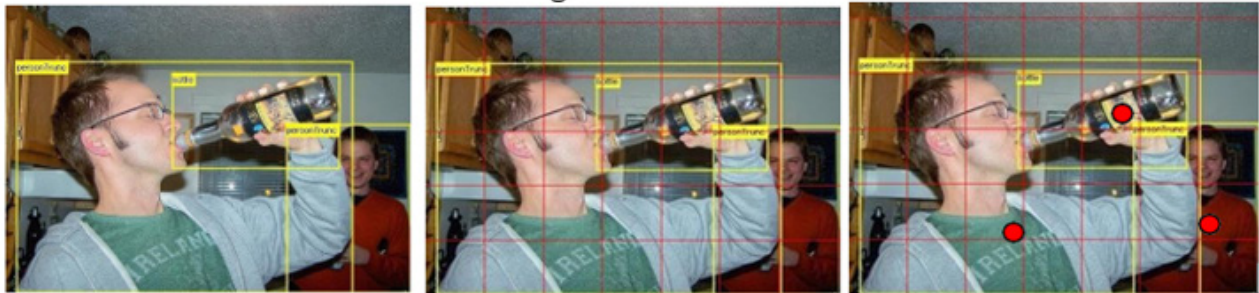


Figure 7: Steps of Modifying Image.

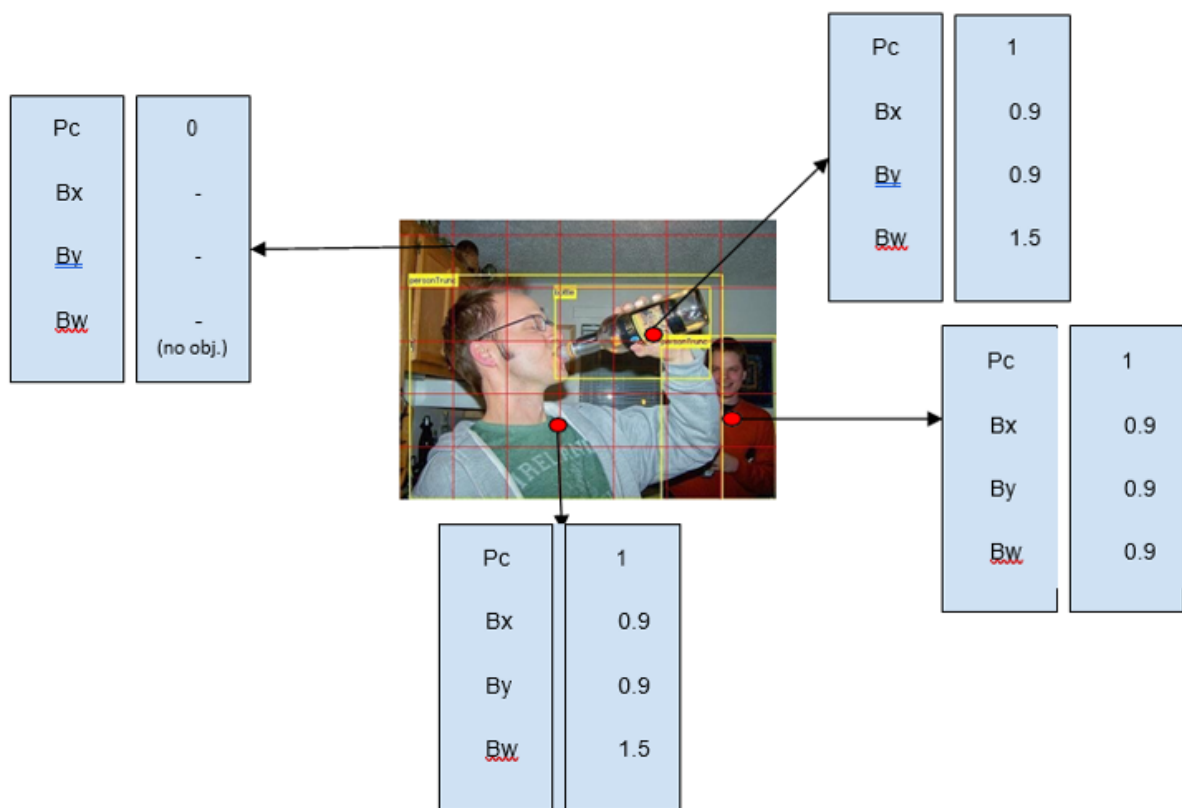


Figure 8: All of the Cells in the Neural Network.

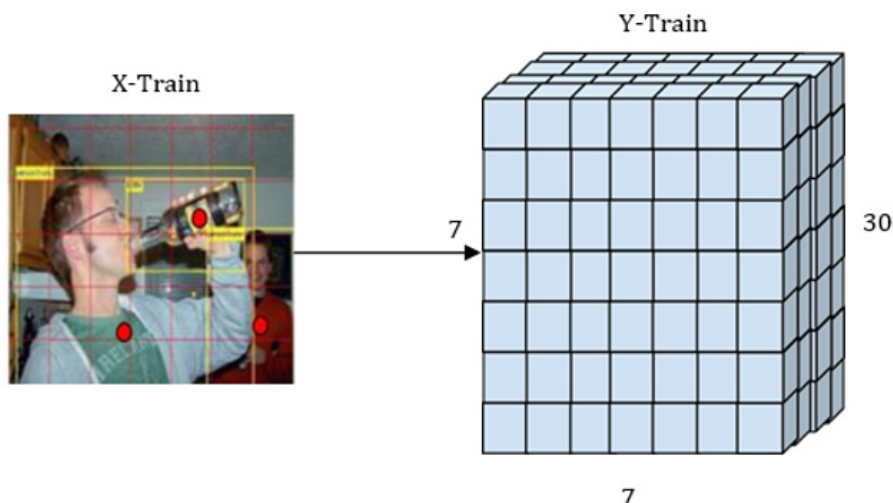


Figure 9: Final Stack of All Tensors in the Neural Network.

## YOLOv2

YOLO9000 or YOLOv2 was published in 2017 by the same authors Joseph Redmon and Ali Farhadi. It included several improvements over the original YOLO. It could also now identify a total of 9000 classes [18]. Some other changes included Batch normalization on Convolutional Layers. Higher Resolution Model training and reduction of spatial sub-sampling which overall improved picking up more detail in images. Removal of all dense layers and making YOLO fully convolutional. Multi-Scale training which allowed YOLOv2 to be more robust at picking up different size objects than its predecessor. Finally adding Anchor boxes which are predefined shapes that match typical shapes of objects. Overall, it reached an average precision of 78.6%, more than 15% improvement on 63.4% from YOLOv1[18].

## Yolov3

Yolov3 was published in 2018, and it also brought many improved and new features to the YOLO family. It first did a major update to its architecture using the Darknet-53 backbone while previous YOLOs used Darknet-19. Bounding box Predictions were changed to using objectiveness scores allowing for more flexibility for misclassified objects. They replaced soft-max classification with binary cross entropy allowing for one object to be multiple classes. Overall, the added layers and changes allow YOLO to increase in accuracy, and overall flexibility.

## YOLOv3-Tiny

YOLOv3-tiny was a smaller alternative that was created by Joseph Redmon [20]. Its main function is to run on constrained environments. It achieves this by having a reduced number of convolutional layers while increasing pooling. YOLOv3-Tiny is able to be ~>400% faster than other models. This, however, comes at the cost of reduced detection having ~15% reduced mAp value [21].

## Yolov4-tiny

YOLOv4-tiny is the smaller brother of YOLOv4. This YOLO model was originally developed by a different team of researchers [22]. They tried experimenting with different features through the research calling these extra features “bag of freebies” and “bag of specials”. It includes Genetic Algorithm training in which they alter the learning rate for the first 10% of training, and self-adversarial training which allows for a more robust system. Once again, Any YOLO-tiny model helps run in constrained environments.

## Yolov7-tiny

YOLOv7-tiny is one of the newest YOLO models covered. Originally it is based off of the YOLOv7 architecture [23]. It has its own “bag of freebies” which includes Planned re-parameterized convolution, Coarse label assignment, Exponential moving averages, and Implicit knowledge. The architecture of this model is concatenation based which reduces hardware usage.

## MTCNN for Human Face Detection

### Face Detection Introduction fps and accuracy of MTCNN and YOLO

The first stage of our robotic vision system is to notice human bodies and then, faces. There are two routes for this task, Face Recognition or Face Detection using Neural Networks. For Face recognition the robots’ sensors can pick up specific human beings from the environment. For example, Facial Recognition allows a sensor to detect a specific person in a crowd, for example a specific celebrity. Face detection allows us to pick up all faces in a crowd that is human. For our robot we need it to run detection, it is not important to know specific faces which will drastically reduce processing power. Modern choices for face detection include MTCNN, Retina Face, and SSD. All of these technologies leverage Convolutional Neural Networks that help make highly accurate detections possible. However, Archi-

texture and layers vary for all these Detection choices.

### Multi-staged Cascade Neural Networks (MTCNN)

Accurate Face Detection and alignment in realistic scenarios is hard to achieve. Many scenarios of lighting, angles, or poses add noise that make accuracy drop substantially. One more problem includes aligning faces which add extra load on top of detection. MTCNN proposes a solution in which they use Multi staged Cascade Neural Networks [16]. Using three levels of Convolved Neural Network (CNNs) to refine candidate windows until the final stage in which it decides on the location of a face, and its landmarks. This leads to highly accurate detections with alignment using landmarks in multiple contexts while reducing overhead compared to other models. With this added accuracy MTCNN is used underneath many existing Facial Recognition models.

### MTCNN Architecture (P-R-O Network structures)

MTCNN [16] has three levels named P-Net (Proposal), R-Net (Refine), and output O-Net. Each new level has a more complex Neural Network.

#### P-Net:

Figure 10 P-Net is the only Fully Convolutional Neural Network [24] and on average is around ~10 layers depending on implementation. The input image is resized and has an image pyramid assigned to itself. This image Pyramid allows for MTCNN to identify human faces in multiple scales and is usually reduced by half or increased, and usually has 5 different scales. Allowing for increased flexibility. The Image Pyramid is fed into the P-Net initializ- es a 12x12 kernel on the right corner. This kernel then feeds these 12x12 windows to P-Net using a feed layer. Figure11.

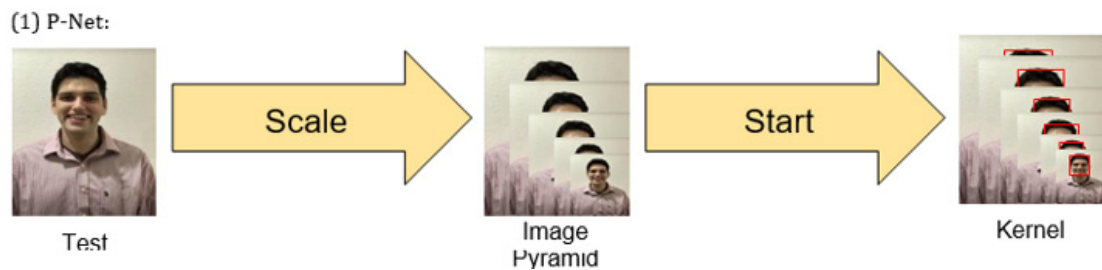


Figure 10: Image Pyramid and Kernel Visualization: MTCNN Architecture containing P-Net.

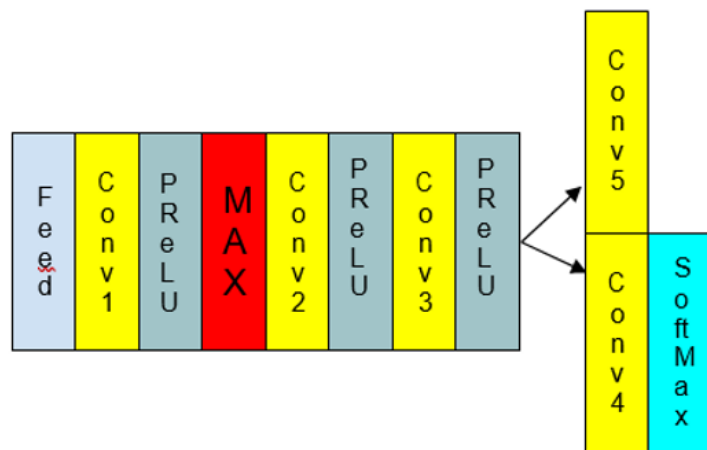


Figure 11: General Layers & Architecture of Proposal Net.



P-Net in general has 5 layers of total Convolutions with extra layers such as Feed, PreLu (Softmax), and MaxPool to regularize and reduce input. From this feed a convolution (Conv1) is run which allows for the CNN to pick out features trained by weights. Then it goes through a PReLU layer which allows the network to pay atten-

tion to features that are important. Added benefits of these layers is it helps solve vanishing gradients in which neurons would stop activating due to ReLU reliance on 0, adding accuracy that comes at a marginal computation cost [25].

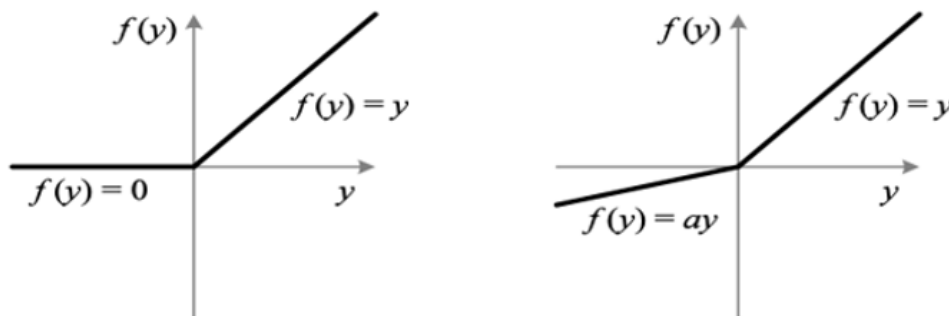


Figure 12: PReLU on the right shows an adaptive slope for negative numbers learned during training.

After the activation function the values are further fed into a maxpool layer. This maxpool layer takes the input and reduces it to a smaller size. It does this by taking the max value of a given portion of the image. This allows for reduced computation and picking up the most distinctive features in an image. This data is then run through 2 more pairs of convolutions and PReLU layers (Conv2-3). Then split into two separate convolutions the first (Conv4) calculates if there is a face or not in the given portion of the image then these numbers are fed into a MaxPool layer turning numerical values into probability. The second separate layer calculates the 4 points of the bounding box Convolution (Conv (5)). Post Processing: Once candidates are found, P-Net uses Bounding box Regression vectors to calibrate candidates. Non-Maximum Suppression (NMS) is then applied to remove overlapping candidate boxes. The specific NMS MTCNN uses is Greedy NMS [24, 26]. This algorithm picks the highest scoring box candidate and depending on an over-

lap, threshold removes boxes that are not needed.

**R-Net:**

From P-net these candidates are fed into R-Net. R-Net’s structure is very similar to P-net, however there are some major differences in its architecture. On average R-Net has 10~20 layers, this allows for more intricate patterns to be seen. R-Net also has a fully connected layer before its final layer to allow for more intricate feature extraction and see “global” context allowing for better classification. Many of the previous final convolutions also become Fully Connected allowing for accurate Feature Extractions. Depending on the MTCNN implementation this stage can have landmark localization. However, in Figure 13 we chose another alternative in which only face Classification and bounding box Classification is implemented.

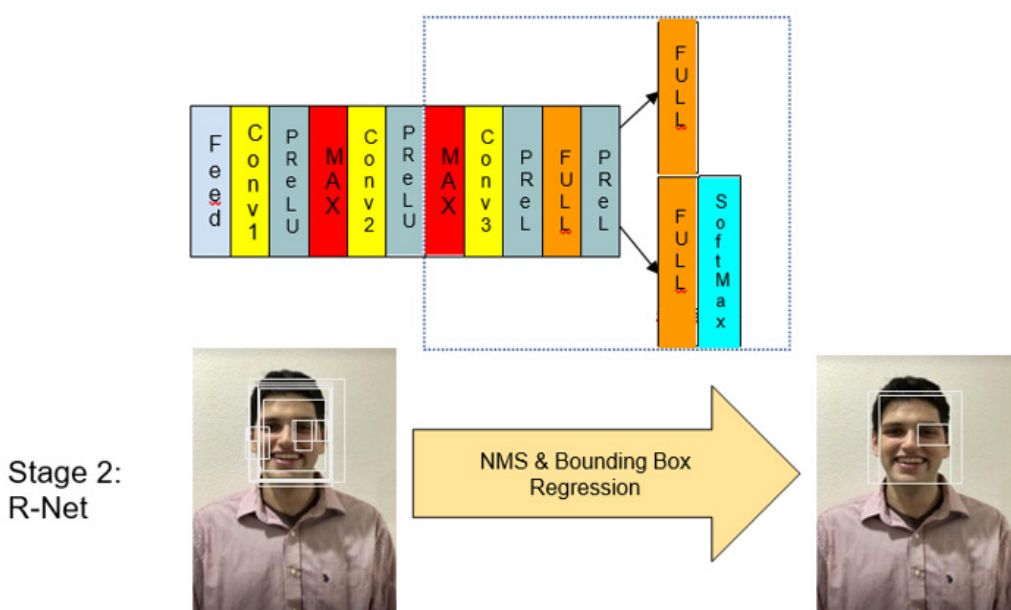


Figure 13: General Layers & Architecture of PR- Net.

**O-Net:**

Figure 14 Candidates from R-Net are fed into O-Net. This architecture is similar to R-Net, however with two major differences. More layers on average 15~25 depending on the implementation goals. This makes O-Net the most complex stage out of the three

stages and allows O-net to always add Facial Landmarks allowing for 5 landmarks: 2 eye points, 2 mouth points at the edge of the lips, and one nose point as shown in the white bounding box in the bottom right Figure 14. Post Processing: Another layer of bounding box regression is applied, and NMS.

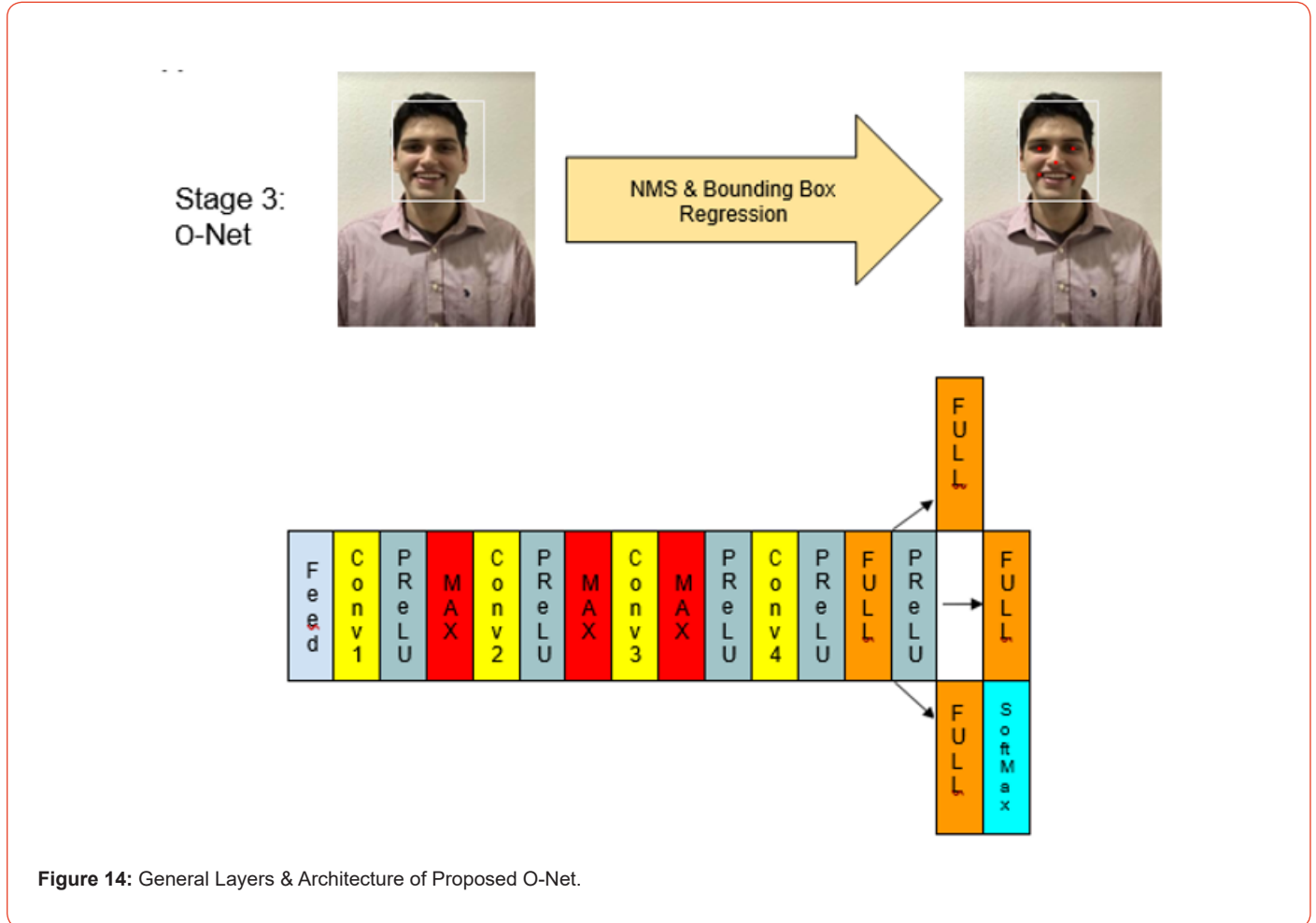


Figure 14: General Layers & Architecture of Proposed O-Net.

**MTCNN Training and Evaluation**

The MTCNN system is trained on three types of Tasks:

i. **Face Classification:** Since this is a Two Classification Problem Cross-Entropy Loss is used [2]:

$$L_i^{det} = -\left(y_i^{det} \log(p_i) + (1 - y_i^{det})(1 - \log(p_i))\right) \quad (1)$$

where  $P_i$  is the probability produced by the network that indicates a sample being a face. The  $y_i^{det}$  notation denotes the ground-truth label. This cross-entropy formula goal is to get as close to 0 as possible.

ii. **Bounding Box Regression:** Each window predicts an offset between it and the nearest ground truth.

$$L_i^{box} = \|\hat{y}_i^{box} - y_i^{box}\|_2^2 \quad (2)$$

where  $\hat{y}_i^{box}$  stands for ground truth and  $y_i^{box}$  stands for regression target There are four coordinates left, top, height, and width. Thus  $y_i^{box} \in \mathbb{R}^4$ .

iii. **Facial Landmark Localization:** Each Facial landmark predicts an offset between itself and the nearest ground truth.

$$L_i^{landmark} = \|\hat{y}_i^{landmark} - y_i^{landmark}\|_2^2 \quad (3)$$

where  $\hat{y}_i^{landmark}$  stands for ground truth and  $y_i^{landmark}$  stands for re-

gression target. There are a total of 10 coordinates for this regres-

sion thus flip is needed;  $\mathcal{Y}_i^{\wedge \text{Landmark}}$  &  $\mathcal{Y}_i^{\text{Landmark}} \in \mathbb{R}^{10}$ . Each coordinate is x, y for the 5 facial locations. These locations as said in sec. 3.3 are the edges of the mouth, eyes, and nose shown on the top-right image in Figure 14.

## Results

### Simulation

These previous detectors were then run on the Jetson Nano, we kept track of a couple of variables. The first being Floating

**Table 1:** Numerical Results of Machine Learning running over the Original Datasets.

Simulation:	BFLOPS (10 <sup>6</sup> )	mAP	AP@Perso	AP@Dog	AP@Car		
MTCNN*	~40	85.7%*	-	-	-	-	FDDB
YOLOv1*	40.155	63%*	-	-	-	Darknet	VOC 2007+2012
YOLOv2	29.371	76.88%	80.24%	89.29%	85.86%	Darknet	VOC 2007+2012
YOLOv3	65.879	54.16%	72.07%	79.86%	60.60%	Darknet	COCO
YOLOv3-Tiny	5.571	17.89%	18.44%	33.17%	7.84%	Darknet	COCO
<b>YOLOv4- Tiny</b>	<b>6.91</b>	<b>48.95%</b>	<b>60.47%</b>	<b>75.16%</b>	<b>46.33%</b>	<b>Darknet</b>	<b>COCO</b>
YOLOv7-Tiny	5.802	55.88%	68.02%	79.67	54.48%	Darknet	COCO

**Table 2:** Experiments for pre-recorded Video for Detection and Recognition.

Detector	FPS	Framework	DataSet
MTCNN	5 FPS	-	WIDER Face/FDDB
YOLOv1	2 FPS	Darknet	VOC 2007+2010
YOLOv2	4.5 FPS	Darknet	VOC 2007+2011
YOLOv3	2.2 FPS	Darknet	VOC 2007+2012
YOLOv3-Tiny	17 FPS	Darknet	COCO
<b>YOLOv4-Tiny</b>	<b>15.4 FPS</b>	<b>Darknet</b>	<b>COCO</b>
YOLOv7-Tiny	12.9 FPS	Darknet	COCO

### Experiments

For the experiment we executed the model on a recorded livestream of our camera in 640x360 pixel webm format. We used the same tools in Simulation to make this possible. We found that YOLOv4-tiny ran the best on the Jetson Nano with an FPS of 15.4 in table 2

### Conclusion

The design of morality and its implementation by AI-enabled autonomous robot that can detect human body detection and further human face detection are demonstrated in this paper. The morally conditioned robot design is not only an offering for scientific inquiry, but also a promising endeavor of future fields as the first step towards testing the morality of autonomous vehicle systems where the responsibility gaps are huge problems in the robotics and AI field. Morally enabled robots may understand moral cognition based on suggested moral rulebook mechanism and further feedback the level of morality via "bumper theory" as the measure

Point Operation a Second (FLOPS) which shows performance of the model on the Jetson Platform, mean Average Precision (mAP) to see robustness of the object detector, Average Precision AP of hand-picked classes that will be used in future research on morality these classes; Person, Dog, and Car. The simulation used the Darknet Framework for YOLO. For MTCNN we used an implementation on Github made for the Jetson Nano [27]. For the simulation we ran the model with its original weights on its chosen dataset to collect our points of interest, we mainly leveraged the tools mentioned before to get these figures. We found that YOLOv7-tiny is optimal because of its higher accuracy, and lower FLOPS, however YOLOv4-tiny would come in a close second [28].

of level of morality to be responsible when there are responsibility gaps. We are hoping that this paper illustrates the importance of addressing the ongoing controversy surrounding robots and further AI morality. While this proposed scientific inquiry cannot single-handedly resolve the responsibility gap issue, we at least want to urge the necessity to step away from the theoretical lens of the issue, in exchange for application. We believe the novel and simple bumper theory approach via moral conditioning will help robots serve for eventually understanding how robots can make moral decisions. We proposed two machine learning vision approaches (YOLO and MTCNN), and we found that YOLOv4-Tiny is the best for our cases in general detection. We are also hoping for these ongoing dialogues and experiments by autonomous robots that may reach out to other scientific endeavors, all of which are in pursuit of making robots responsible as moral agents.

### Acknowledgement

The authors deeply appreciate Patrick R. Audije and Hoang And Pham as the graduate students in SOECS at UOP who have dis-

cussed together autonomous mobile robotics with machine intelligence especially, deep learning with computer vision and multiple single boards under various platforms with Dr. Lee through seminars or workshops how to implement such morally responsible robots and helped insights about the technological development and experiments.

### Conflict of interest

No conflict of interest.

### References

1. Redmon S, Divvala R, Girshick, A Farhadi (2016) You Only Look Once: Unified, Real-Time Object Detection, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 779-788.
2. MH Putra, ZM Yussof, KC Lim, SI Salim (2018) Convolutional Neural Network for Person and Car Detection using YOLO Framework. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 9: 2-13.
3. Deisy Chaves, Eduardo Fidalgo, Enrique Alegre, Rocío Alaiz-Rodríguez, Francisco Jáñez-Martino et al., (2020) Assessment and Estimation of Face Detection Performance Based on Deep Learning for Forensic Applications. *Sensors (Basel)* 20(16): 4491.
4. YE Bigman, A Waytz, R Alterovitz, K Gray (2019) Holding Robots Responsible: The Elements of Machine Morality. *Trends in Cognitive Sciences* 23(5): 365-368.
5. M Robillard (2017) No Such Thing as Killer Robots, *Journal of Applied Philosophy* 35(4): 705-717.
6. I Taylor (2021) Who Is Responsible for Killer Robots? Autonomous Weapons, Group Agency, and the Military-Industrial Complex. *Journal of Applied Philosophy* 38(2): 320-334.
7. RD Jong Elissa (2020) The Retribution-Gap and Responsibility-Loci Related to Robots and Automated Technologies: A Reply to Nyholm, *Science and Engineering Ethics* 26: 727-735.
8. Q Zhu, T Williams, B Jackson, R Wen (2020) Blame-Laden Moral Rebukes and the Morally Competend Robot: A Confucian Ethical Perspective. *Science and Engineering Ethics* 26: 2511-2526.
9. J Danaher (2020) Welcoming Robots into the Moral Circle: A Defence of Ethical Behaviourism. *Science and Engineering Ethics* 26: 2023-2049.
10. J.P. Sullins (2011) When a Robot a Moral Agent is, in *Machine Ethics*, M. Anderson and S.L. Anderson (Eds.) Massachusetts: Cambridge University Press, pp. 151-161.
11. S. Nichols (2018) The Wrong and the Bad, in *Atlas of Moral Psychology*, K. Gray and J. Graham (Eds.) New York: The Guilford Press, pp. 40-49.
12. J Graham, J Haidt, S Koleva, M Motyl, R Iyer et al., (2013) Moral Foundations Theory: The Pragmatic Validity of Moral Pluralism, *Advances in Experimental Social Psychology* 47: 55-130.
13. RF Baumeister (2018) Free Will and Moral Psychology, in *Atlas of Moral Psychology*, K. Gray and J Graham (Eds.) New York: The Guilford Press, pp. 332-338.
14. B. Brembs (2011) Towards a Scientific Concept of Free Will as a Biological Trait: Spontaneous Actions and Decision-Making in Invertebrates, in *Proceedings of the Royal Society B: Biological Sciences* 238: 930-939.
15. J Voiklis, BF Malle (2018) Moral Cognition and Its Basis in Social Cognition and Social Regulation, in *Atlas of Moral Psychology*, K Gray and J Graham (Eds.) New York: The Guilford Press, pp. 108-121.
16. Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, Yu Qiao (2016) Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks, *IEEE Signal Processing Letters (SPL)* 23(10): 1499-1503.
17. J Joseph Redmon, Ali Farhadi (2016) YOLO9000: Better, Faster, Stronger, *Computer Vision and Pattern Recognition*.
18. Joseph Redmon and Ali Farhadi (2018) YOLOv3: An Incremental Improvement, *CVPR*.
19. Juan R, Terven and Diana M, Cordova Esparza (2023) A Comprehensive Review Of YOLO: From YOLOv1 and Beyond.
20. Adarsh (2020) YOLO v3-Tiny: Object Detection and Recognition using one stage improved model, 6th International Conference on Advanced Computing and Communication Systems (ICACCS).
21. Darknet, Redmon at pjreddie dot com.
22. Alexey Bochkovskiy, Chien Yao Wang, Hong Yuan Mark Liao (2020) YOLOv4: Optimal Speed and Accuracy of Object Detection.
23. Alexey Bochkovskiy, Chien Yao Wang, Hong Yuan Mark Liao (2022) YOLOv7: Optimal Speed and Accuracy of Object Detection.
24. Long Hua Ma, Hang Yu Fan, Zhe Ming Lu, Dong Tian (2020) Acceleration of multi-task cascaded convolutional networks. *IS&T Image Processing Journal* 14(11): 2435-2441.
25. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun (2015) Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, *International Conference on Computer Vision (ICCV)*.
26. Jan Hosang, Rodrigo Benenson (2017) Learning non-maximum suppression Openaccess, *CVPR*.
27. MTCNN, jkjun-avt TensorRT Demos at Github.
28. Long Hua Ma, Hang Yu Fan, Zhe Ming Lu, Dong Tian (2020) IET Image Processing Research Article Acceleration of multi-task cascaded convolutional networks. *IET Image Processing Journal* 14(11): 2435-2441.