Adaptive Navigation for the Visually Impaired: Safe Reinforcement Learning with Real-Time Computer Vision

by

Sharif Mohammad Omar Faruq
22101429
Wasif Khan
22101357
Saif Alam Shaan
22101368

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
BRAC University
October 2025.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Sharif Mohammad Omar Faruq

22101429

Wasif Khan

22101357

Saif Alam Shaan

22101368

Approval

The thesis/project titled "Adaptive Navigation for the Visually Impaired: Safe Reinforcement Learning with Real-Time Computer Vision" submitted by

1. Sharif Mohammad Omar Faruq (22101429)
2. Wasif Khan (22101357)
3. Saif Alam Shaan (22101368)

of Summer, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science in Summer, 25 Year.

Examining Committee:

Supervisor:
(Member)

Md. Golam Rabiul Alam

Professor
Dept. of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Md. Ashraful Alam

Associate Professor
Dept. of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Md. Golam Rabiul Alam

Professor
Dept. of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi

Chairperson
Department of Computer Science and Engineering
Brac University

Ethics Statement

This research will offer mobility assistance for the visually impaired population, a population that deserves careful ethical consideration. All of our proposed development prioritizes user safety, autonomy, and dignity. Our research design follows established ethical principles for assistive navigation technology. Manduchi and Coughlan[5] emphasize that vision-based navigation systems for blind users must prioritize user agency over automation, ensuring that technology provides informational support rather than replacing human decision-making or replace existing mobility aids. This principle guided our architectural choice to implement advisory collision avoidance rather than autonomous control. We used open-source Datasets Mapilary Vista v2.0, Cityscape, and CARLA simulator version 0.9.16 for simulation. Thus, there was no intervention of privacy during training, and no direct human subject was risked during the simulation. We emphasize that this work represents only foundational research. Real-world deployment would require extensive testing with visually impaired users and collaboration with orientation and mobility specialists.

Abstract

This paper presents a vision-only assistive navigation framework that couples real-time panoptic segmentation with safe reinforcement learning to provide proactive, collision-aware guidance for visually impaired pedestrians in dynamic urban environments using only RGB cameras on resource-constrained devices. The approach integrates a lightweight bottom-up MobileNetV3–FPN panoptic segmentation model for unified scene understanding, a ConvGRU module that predicts short-horizon danger maps from temporal mask sequences, and a multi-input policy that conditions decision-making on both current semantics and anticipated risk. Safety is enforced by a PPO-based controller trained under a constrained formulation (Lagrangian) and supplemented at runtime with an action-shielding safety layer that filters unsafe actions. The system is trained and evaluated in CARLA with domain diversity from Cityscapes and Mapillary Vistas, emphasizing ethical, simulation-first validation and deployment feasibility on edge hardware. Experiments and studies indicate that constrained PPO with action shielding reduces safety violations compared to unconstrained PPO, while ConvGRU-based temporal prediction improves anticipatory avoidance of dynamic obstacles, achieving a favorable speed–accuracy trade-off for wearable use cases with the MobileNetV3 panoptic variant. The work contributes an affordable RGB-only stack, a tightly coupled perception–prediction–control design for proactive safety, and a reproducible benchmark that surfaces limitations in small-object instance quality and sim-to-real transfer, outlining targeted directions for future refinement.

Keywords: assistive navigation, panoptic segmentation, ConvGRU, safe reinforcement learning, action shielding, CARLA, vision-only systems.

Acknowledgement

First and foremost, we would like to thank our Creator, the Almighty Allah, for making our journey possible and removing all obstacles from our paths during our thesis. Secondly, we are grateful to both our thesis supervisors, Professor Dr. Md. Golam Rabiul Alam, sir, and Md. Ashraful Alam sir for their support and advice during the thesis. Both of them shared their wisdom and guidelines whenever we reached out to them without any hesitation. Finally, to our parents, friends, and acquaintances. It wouldn't have been possible for us to continue this journey without their prayer and kind support.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$*$	Convolution operation
δ	Safety threshold for action shielding
ϵ	Label smoothing parameter or clipping parameter in PPO
γ	Discount factor in reinforcement learning
$\hat{A}_t$	Estimated advantage in PPO
λ	Lagrange multiplier for constraint satisfaction
$\mathcal{A}$	Action space in MDP
$\mathcal{S}$	State space in MDP
$\odot$	Element-wise multiplication
π	Policy function in reinforcement learning
σ	Sigmoid activation function
θ	Policy network parameters
$c(s, a)$	Cost function for safety constraints
h_t	Hidden state at time t in GRU
$J(\pi)$	Expected cumulative reward of policy
$r(s, a)$	Reward function
r_t	Reset gate in GRU
z_t	Update gate in GRU
AAL	Ambient Assisted Living
AP	Average Precision
AR	Augmented Reality
ASPP	Atrous Spatial Pyramid Pooling

BCE Binary Cross-Entropy

BEV Bird's-Eye View

CARLA Car Learning to Act

CE Cross-Entropy

CMDP Constrained Markov Decision Process

CNN Convolutional Neural Network

ConvGRU Convolutional Gated Recurrent Unit

CPO Constrained Policy Optimization

CVaR Conditional Value-at-Risk

Dice Dice Similarity Coefficient

DQN Deep Q-Network

DSConv Depthwise Separable Convolution

ECA Efficient Channel Attention

FHWA Federal Highway Administration

FOV Field of View

FPN Feature Pyramid Network

FPS Frames Per Second

GAE Generalized Advantage Estimation

GRU Gated Recurrent Unit

LiDAR Light Detection and Ranging

LSTM Long Short-Term Memory

MDP Markov Decision Process

mIoU Mean Intersection over Union

NMS Non-Maximum Suppression

PET Post-Encroachment Time

PQ Panoptic Quality

RCMDP Robust Constrained Markov Decision Process

RNN Recurrent Neural Network

SLAM Simultaneous Localization and Mapping

SRL Safe Reinforcement Learning

STGRU Spatio-Temporal Transformer Gated Recurrent Unit

TTC Time-to-Collision

WHO World Health Organization

YOLO You Only Look Once

Chapter 1

Introduction

1.1 Background

According to the World Health Organization (WHO), over 2.2 billion people around the world are suffering from severe sight loss or blindness [30], and these numbers will keep getting higher as the population grows older. Because vision impairment is strongly correlated with elderly people. Approximately 80% of the people with blindness or visual impairment are aged 50 or above [47]. According to The American Foundation for the Blind, individuals with low vision have "permanently reduced vision that cannot be corrected with regular glasses, contact lenses, medicine, or surgery." People who can see only a portion of the scene are considered to have partial vision. Good central vision with poor peripheral vision is also considered partial vision. In most scenarios, blind people can see to some extent, and 15% of that population has total blindness, which refers to total darkness [3]. They generally have to rely on using a guide cane or to touch objects physically to sense their location. However, it is very difficult for them to approximate the location of moving objects. They must identify moving objects because they can arrive in their way anytime without them sensing their presence. To be specific, blind pedestrians encounter substantial difficulty in navigating their surroundings, resulting in reduced independence and a greater risk of accidents.

The effects of this limitation are serious. Research shows that blind and visually impaired pedestrians face significantly higher traffic risks than sighted pedestrians. Studies indicate that approximately 50% of blind pedestrians experience at least one traffic-related incident (near-miss or collision) annually, with inadequate auditory cues and inaccessible intersection designs identified as primary contributing factors [4]. According to the World Health Organization, road traffic injuries cause approximately 1.19 million deaths annually, with pedestrians and cyclists comprising 26% of those fatalities [56]. Visually impaired individuals represent a disproportionately vulnerable subgroup within this population, particularly in urban environments lacking accessible infrastructure [56]. These statistics demonstrate that current infrastructure and assistive technologies remain insufficient for ensuring pedestrian safety among blind individuals.

1.2 Rational of the Study

This section discusses the significance and relevance of the research gap and its impact. Vision-impaired individuals are not benefited from the current technological advancement in (AR) and virtual reality (VR). These advancements should be used to assist and guide them in their day-to-day lives. Even though this is a traditional subject that has been studied for many years, research teams are always coming up with new ideas and giving optimism that vision impairment won't be a permanent problem in the future. Assistive technologies aiming at easing travel are often divided into three areas: electronic travel aids, electronic orientation aids, and position locator devices, described by Elmannai et al.[20] as "devices that gather information about the surrounding environment and transfer it to the user," "devices that provide pedestrians with directions in unfamiliar places," and "devices that determine the precise position of their holder," respectively. Attempts have been made by launching some applications in the market, but due to their lack of user-friendly interface and inaccuracy, they were deemed unsuccessful. Other existing models are limited by light source and bulky hardware requirements.

This paper will explore how reinforcement learning and computer vision can improve accessibility for those with visual impairments. Advancements in AI and reinforcement learning can improve mobility and safety for visually impaired people. The purpose of this work is to investigate how safe reinforcement learning can be used to assistive navigation systems, guaranteeing that these technologies prioritize user safety while still aiding navigation.

The need for blind people to benefit from sophisticated gadgets and systems of navigation has led to the efficient and rapid development of brilliant systems for navigation. From statistics compiled by the World View, there are an estimated 650,000 blind adults in Bangladesh. Approximately 1.53% of adults aged 30 and older are bilaterally blind[2]. Many people in this population have difficulty in moving, and more so in the congested and changing urban environment. The traditionally available navigation aids, including canes and guide dogs, are very limited since their efficiency decreases significantly especially in areas which have high traffic density and which require frequent crossing, for instance roads and pedestrian bridges where vehicles, traffic signals, and other pedestrians are constantly in motion. Conventional approaches of guidance and orientation do not offer continuous and/or context-sensitive directions; this is why there is a steady demand for sophisticated techniques which would be able to accommodate the given context of safe and reliable assisted navigation[44].

In principle, visually impaired people need various types of real-time data for navigation, for example, object detection (pedestrians, cars, traffic signals), distance measurement for the identification of the obstacles, scene understanding such as the identification of road signs, or curbs. Though these data types are critical for navigation assistance their collection and processing as well raises concern for user privacy and security. To guarantee real-time response to the user, many assistive technologies require capturing context-sensitive data which may make users vulnerable to hackers or stalkers when such technologies capture environmental information or user's preferences[18].

Safe Reinforcement Learning (SRL) seems to be a viable solution addressing these challenges. SRL is a type of machine learning that enables different systems to make choices, but at the same time remain safe, which is perfect for applications inherent in modern urban streets. Thus, on the basis of finding commonalities with the environment, SRL can improve navigation paths and minimize the risks affecting the user[45]. At the same time, real-time navigation heavily relies on Computer Vision (CV) to identify objects, measure distances and recognize complex scenes. Some of the modern types of CV models such as You Only Look Once (YOLO) and Single Shot MultiBox Detector (SSD) work well in matters concerning object detection and distance estimation which is essential for navigation[39]. Despite these improvements navigation systems still have drawbacks - the following are the drawbacks of existing navigation systems. Most of the conventional assistants that are based on smartphone applications are those that necessarily rely on some prior physical installation or system, or else on preprogrammed data, which makes them less efficient if not always entirely unusable in the context of constantly evolving metropolitan environments. Although models based on SRL are flexible, the approaches based on these models are not suitable for the fast response to changing situations. CV-based systems, however, encounter problems with varying lighting conditions and frequently fail to perform data processing in real time[40].

1.3 Problem Statement

Blind and visually impaired individuals face life-threatening challenges when navigating through busy urban streets filled with moving vehicles, uneven pavements, and unpredictable pedestrians. Traditional aids such as white canes and guide dogs provide limited spatial awareness, detecting only nearby obstacles and offering no information about fast-moving dangers like cars or bicycles. Although GPS-based navigation apps can guide users to a destination, they fail to interpret immediate surroundings or detect dynamic obstacles, leaving users vulnerable during tasks such as crossing roads or navigating crowded areas. These limitations often lead to near-collision incidents or even fatal accidents, severely restricting the independence and safety of visually impaired pedestrians.

While several smart technologies have been developed to improve navigation for the blind, they remain far from practical or inclusive. Many rely on pre-mapped environments or costly sensor-based hardware that is unaffordable for most users, especially in developing countries. Moreover, such systems often fail to function effectively in changing lighting or traffic conditions, and they cannot interpret complex, real-time urban scenarios. As a result, the existing solutions still fall short of providing a reliable, context-aware, and affordable means of safe navigation—leaving millions of visually impaired individuals exposed to daily risks in modern city environments. This research asks a critical question:

"Can we combine Safe Reinforcement Learning with real-time Computer Vision using only cheap cameras to create a system that predicts dangerous situations before they happen, keeps users safe through guaranteed collision prevention, and costs little enough for everyone to use?"

1.4 Objective

The primary objective of this study is to design a vision-only safe navigation system for visually impaired individuals using Safe Reinforcement Learning (SRL) and Real-Time Computer Vision. The goal is to create a framework capable of understanding, predicting, and avoiding potential hazards in dynamic urban environments, ensuring both safety and autonomy.

- To design a lightweight paoptic segmentation network using MobileNetV3 and bottom up approach that can identify critical classes such as roads, vehicles, and pedestrians from RGB camera input. This visual understanding forms the foundation for safe navigation without relying on LiDAR, GPS, or depth sensors.
- To implement a GRU-based temporal prediction mechanism capable of forecasting the future positions of dynamic obstacles. By predicting danger-prone areas a few frames ahead, the agent gains short-term foresight, allowing safer and smoother movement in changing environments.
- To combine the current semantic map with the predicted danger channel, forming a multi-channel input that enables the agent to reason about both present and upcoming risks. This fusion enhances the spatial and temporal awareness required for safe real-time decision-making.
- To train a Proximal Policy Optimization (PPO) agent augmented with a safety layer that sends 4 discrete actions (left, right, forward, stop) and prevents unsafe actions during learning and deployment. The agent will be trained in the CARLA simulator, ensuring adaptive and collision-free navigation in complex traffic conditions.
- To assess the trained model's performance using metrics such as collision rate, path efficiency, inference time, and generalization capability. The results will validate whether the system can ensure reliable, real-time navigation and set a benchmark for future assistive navigation technologies.

1.5 Methodology in Brief

This study adopts a simulation-based experimental design to evaluate the effectiveness of a vision-only navigation system for visually impaired individuals. The research integrates computer vision for real-time perception and safe reinforcement learning for decision-making in dynamic urban environments, allowing controlled yet realistic assessment of navigation performance.

The system's perception is built using MobileNetV3 with DeepLabV3 to generate semantic segmentation maps of the environment. A GRU-based temporal prediction module forecasts future positions of moving obstacles, producing a predicted danger map. These outputs are fused into a multi-channel representation that serves as input to a PPO agent augmented with a safety layer, ensuring collision-free navigation while respecting predefined safety constraints such as maintaining minimum distances from pedestrians, vehicles, and obstacles.

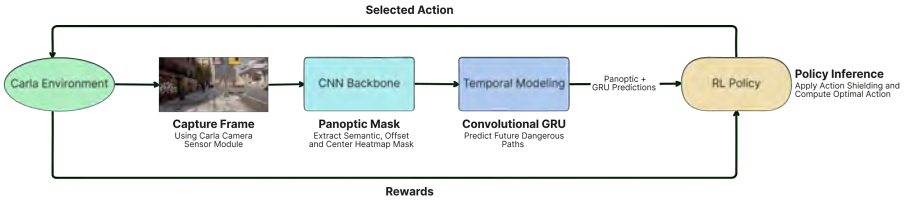


Figure 1.1: System Architecture.

Quantitative evaluation focuses on both perception and navigation metrics. Object detection performance is measured using mean Average Precision (mAP@0.5), while reinforcement learning performance is assessed via loss function, collision rate, path efficiency, and average reward per episode, providing insights into navigation accuracy, lagrangian cost, and adaptability in dynamic scenarios. The methodology ensures risk-free testing in simulation, maintaining ethical standards while enabling realistic evaluation of the system's capabilities.

1.6 Scopes and Challenges

The scope of this study focuses on developing a vision-based safe reinforcement learning (SRL) navigation system that provides real-time feedback to users, particularly individuals with visual impairments, enabling safe and effective navigation in dynamic environments. The system integrates MobileNet-based computer vision (CV) models for detecting both static and moving obstacles, and leverages ConvGRU networks to predict the temporal dynamics of moving objects. Feedback is delivered through auditory and haptic channels, and the system prioritizes data privacy and security, ensuring user information is protected during operation.

The approach combines SRL and CV methodologies: the SRL agent learns safe navigation policies in complex urban environments with dynamic obstacles, while MobileNet-based CV models are trained on large datasets to detect and estimate distances to objects under varying lighting and weather conditions. The ConvGRU predicts near-future states of moving obstacles, enabling proactive avoidance. Post-training, the system will be evaluated in real-world scenarios to assess its efficiency, adaptability, and safety.

Several challenges are anticipated. Real-time processing of high-dimensional visual and temporal data while maintaining low latency is computationally demanding. Dynamic environments, including moving crowds, traffic, and lighting variations, may reduce detection and prediction accuracy. Ensuring robustness and generalization across diverse urban settings remains difficult. Maintaining user privacy and data security, along with providing accurate and timely feedback through SRL and ConvGRU predictions, is critical for safe, practical deployment.

Chapter 2

Literature Review

2.1 Preliminaries

Developing a vision-based safe navigation system for autonomous agents requires a combination of perception, decision-making, and predictive modeling. This section introduces the fundamental concepts that enable such a system to operate effectively in dynamic, real-world environments. It begins with techniques for understanding complex visual scenes, followed by methods for learning optimal behaviors through interaction. Special attention is given to strategies that ensure safety during learning and execution, as well as approaches for predicting future states of dynamic elements in the environment. Finally, the simulation platform that allows safe training and evaluation of these methods is presented. Together, these foundations provide the theoretical and practical basis for the design, training, and validation of a safe, vision-guided navigation agent.

2.1.1 Image Segmentatio

Semantic segmentation involves classifying each pixel in an image into a predefined category, such as road, sky, or pedestrian. While effective for understanding the general layout of a scene, it treats all objects of the same class identically, lacking differentiation between individual instances. On the other hand, instance segmentation distinguishes between separate *thing* class (countable objects). For example, it can differentiate between two cars in the same category. But, instance segmentation can not categorize *stuff* (backgrounds) class objects. However, panoptic segmentation combines the strengths of both semantic and instance segmentation. It assigns a unique label to every pixel, distinguishing between different instances of *thing* class and categorizing all objects (e.g., stuffs, things) into semantic classes. This unified approach offers a comprehensive understanding of a scene, making it particularly useful for applications requiring detailed scene interpretation.

Furthermore, a bottom-up approach in computer vision starts from low-level elements, like pixels and edges, and gradually builds up to recognize objects and scenes. Unlike top-down methods that rely on predefined models, bottom-up approaches learn directly from the raw visual data. This is especially important in dynamic and unpredictable environments, allowing systems to adapt in real time. In panoptic segmentation, bottom-up strategies enable faster processing and greater

flexibility compared to computationally intensive top-down methods. For example, Panoptic-DeepLab uses a bottom-up approach to achieve state-of-the-art results on datasets like Cityscapes and Mapillary Vistas [28].

2.1.2 Temporal Model With Gated Recurrent Unit (GRU)

A Gated Recurrent Unit (GRU) is a recurrent neural network architecture designed to model temporal dependencies in sequential data while mitigating the vanishing gradient problem that plagues standard RNNs. Introduced by Cho et al. [9], GRUs employ two gating mechanisms—the **update gate** z_t and **reset gate** r_t—to regulate information flow across time steps. Mathematically, given input x_t and previous hidden state h_{t-1}, the GRU computes:

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \tag{2.1}$$

for the reset gate, which determines how much past information to discard;

$$\tilde{h}_t = \tanh(W_h x_t + U_h(r_t \odot h_{t-1})) \tag{2.2}$$

for the candidate hidden state;

$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \tag{2.3}$$

for the update gate, which balances retention of previous state versus incorporation of new information; and finally

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{2.4}$$

for the updated hidden state, where σ denotes the sigmoid function, $\odot$ represents element-wise multiplication, and W, U are learnable weight matrices. This architecture requires fewer parameters than Long Short-Term Memory (LSTM) networks while maintaining comparable performance on sequence modeling tasks.
For vision-based navigation tasks, **Convolutional GRUs (ConvGRUs)** extend standard GRUs by replacing fully connected operations with convolutional layers, enabling spatial feature preservation across temporal sequences. In ConvGRU, all weight matrices W and U become 2D convolutional kernels (W_z, W_r, W_h and U_z, U_r, U_h), and operations are performed via spatial convolutions ($*$) rather than matrix multiplications:

$$z_t = \sigma(W_z * x_t + U_z * h_{t-1}) \quad \text{and} \quad r_t = \sigma(W_r * x_t + U_r * h_{t-1}) \tag{2.5}$$

This formulation allows the network to process sequences of spatial feature maps—such as semantic segmentation masks or BEV representations—while maintaining spatial structure and sharing parameters across pixel locations. ConvGRUs have demonstrated superior efficiency over 3D CNNs and Transformers for capturing long-range temporal dependencies in video segmentation tasks, as they accumulate temporal context in recurrent hidden states rather than requiring prohibitively deep architectures or attention mechanisms with quadratic complexity. For autonomous navigation systems processing temporal sequences of semantic masks or obstacle detections, ConvGRUs provide an optimal balance between computational efficiency (lower memory footprint than LSTMs, faster than Transformers) and representational capacity for modeling dynamic scene evolution over extended time horizons.

2.1.3 Reinforcement Learning and Safety

Reinforcement Learning (RL) is a framework where an agent learns to make sequential decisions by interacting with an environment, mathematically modeled as a *Markov Decision Process (MDP)*. An MDP is defined as the tuple $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$, where $\mathcal{S}$ represents the state space, $\mathcal{A}$ the action space, $P(s'|s,a)$ the transition probability, $r(s,a)$ the reward function, and $\gamma \in [0,1)$ the discount factor [27]. The goal is to find a policy $\pi(a|s)$ that maximizes the expected cumulative reward:

$$J(\pi) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right]. \tag{2.6}$$

When the state or action space becomes **large or continuous**, representing value functions using a discrete state-action table becomes infeasible. To handle such high-dimensional spaces, **neural networks (NNs)** are employed as function approximators for value functions or policy mappings, enabling RL to scale to complex real-world problems such as autonomous driving or visual navigation. However, this scalability introduces new safety challenges, as uncontrolled exploration in continuous environments can lead to dangerous or irreversible outcomes during both training and deployment.

To address safety concerns, RL is extended into the *Constrained Markov Decision Process (CMDP)* framework, defined as $(\mathcal{S}, \mathcal{A}, P, r, c, d, \gamma)$, where $c(s,a)$ is a cost function representing unsafe behaviors, and d denotes the maximum allowable expected cost [13]. The CMDP optimization problem is formulated as:

$$\max_\pi J(\pi) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r_t \right], \quad \text{s.t. } \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t c_t \right] \leq d. \tag{2.7}$$

Through **Lagrangian relaxation**, a multiplier $\lambda > 0$ transforms the constrained problem into:

$$\mathcal{L}(\pi, \lambda) = J(\pi) - \lambda(\mathbb{E}_\pi[C] - d), \tag{2.8}$$

allowing gradient-based optimization while maintaining constraint satisfaction. This formulation underpins safe policy gradient methods such as **Proximal Policy Optimization (PPO)** [22]. PPO improves stability by introducing a clipped surrogate objective that prevents large policy updates:

$$L^{\text{CLIP}}(\theta) = \mathbb{E}_t \Big[\min(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t) \Big], \tag{2.9}$$

where $r_t(\theta)$ is the policy probability ratio between new and old policies, and $\hat{A}_t$ is the estimated advantage. By constraining policy changes, PPO achieves a balance between exploration and stability, making it widely used in safety-critical applications.

Building upon the CMDP foundation, **Safe Reinforcement Learning (Safe RL)** introduces mechanisms to ensure safety during both exploration and execution. Rather than penalizing unsafe actions after they occur, Safe RL proactively prevents constraint violations in real time [23]. A notable method is **Action Shielding**, where a safety layer evaluates each candidate action a_t through a risk function $S(s_t, a_t)$. If $S(s_t, a_t)$ exceeds a predefined threshold δ, the action is replaced with a safer alternative:

$$a'_t = \arg\min_{a \in \mathcal{A}} S(s_t, a). \tag{2.10}$$

This results in a **safe policy** expressed as:

$$\pi_{\text{safe}}(a|s) = \pi(a|s)\mathbb{I}[S(s,a) < \delta], \tag{2.11}$$

ensuring that only safe actions are executed. Such proactive safety mechanisms enable RL agents to operate reliably in real-world domains, including autonomous driving, robotic manipulation, and assistive navigation for visually impaired individuals.

2.1.4 Carla Simulator

CARLA (Car Learning to Act) is an open-source urban driving simulator designed for autonomous driving research, supporting development, training, and validation of driving agents in realistic, flexible environments [19]. It provides a wide range of functionalities including configurable sensor suites (RGB, depth, semantic segmentation, LiDAR, etc.), and supports dynamic environmental conditions such as weather changes, time of day/lighting variation, and road surface changes. The simulator also models traffic with multiple vehicle behaviors (traffic lights, obeying rules, lane changes) and crowd/pedestrian agents with AI-driven walking behaviors, enabling interactions with moving agents in urban scenes.
Because CARLA allows full control over scenario complexity, it is ideal for studying vision-based navigation under varying conditions, including dynamic obstacles, occlusions, and environmental perturbations. In your work, CARLA's support for dynamic weather & lighting, AI pedestrian crowd simulation, and traffic generation gives you a robust testbed for safe RL navigation under realistic constraints.

2.2 Review of Existing Research

This part aims to focus on the past relevant work done in the field of assistive living for individuals with visual impairment in the context of Safe Reinforcement Learning and Computer Vision. Different frameworks and models have been analyzed and discussed how a visually impaired individual could benefit from it while navigating in an urban area. This review will further critically assess the limitation of the current approaches in order to improve on those shortcomings.

The research paper [36] discusses a system utilizing deep reinforcement learning (DRL) combined with ultrawide-bandwidth (UWB) beacons and voice feedback to help avoid obstacles and provides feedback to an intuitive interface to the visually impaired people. This system focuses on both indoor and dynamic pedestrian environments which uses SLAM (Simultaneous Localization and Mapping) framework to help the robot determine its path and navigate safely. The UWB beacons are installed for a path and helps the robot to calculate its location with high accuracy to determine the most optimal way to navigate through checkpoints. Despite the DRL model learning how to navigate through dynamic environments, UWB beacons must be pre-installed in the environment, which makes the system less adaptive to new or unfamiliar environments. Furthermore, this paper mentions dynamic human obstacles but does not account for large scale environments which includes dynamic cars, road signs, traffic signal and so on.

The pursuit of effective Electronic Travel Aids (ETAs) for the visually impaired began with foundational systems focused on the primary challenge of collision avoidance. Early work, such as the wearable obstacle detection system developed by Benjamin, Ali, and Schepis (2001), represents this initial paradigm. Their approach utilized infrared sensors to detect the presence and proximity of objects in the user's path, translating this data into tactile or auditory feedback. While pioneering in its effort to provide real-time environmental information, the scope of such systems was limited to rudimentary obstacle detection [1]. They could alert a user to an impending object but lacked the ability to provide any semantic context, leaving the high cognitive load of interpreting the environment and making complex navigational decisions entirely on the user.

Over the past two decades, the field has been transformed by advancements in artificial intelligence and computer vision. A comprehensive review by [41] captures this technological shift, detailing the transition from basic sensory substitution devices to sophisticated, intelligent systems. The authors highlight the pivotal role of deep learning, particularly Convolutional Neural Networks (CNNs), in enabling modern ETAs to perform complex perceptual tasks. These include not just obstacle detection but also semantic segmentation for understanding different terrain types, object recognition to identify key environmental features like crosswalks and doors, and monocular depth estimation to build a 3D understanding of the scene from a single camera. This body of work establishes the current technological foundation, demonstrating that deep learning provides the necessary tools for rich, real-time environmental interpretation.

Building directly on these deep learning foundations, the system proposed by [35] offers a practical implementation of a modern ETA. Their device functions as a "smart cane" that integrates real-time object detection with depth information to provide users with intuitive and context-aware guidance. By leveraging pre-trained models, the system can identify a wide array of common urban obstacles and landmarks, such as vehicles, pedestrians, and staircases, and communicate this information effectively to the user. This work serves as a prime example of a semantically aware navigation aid. However, its decision-making logic remains fundamentally rule-based; it reacts to detected objects according to a fixed set of pre-programmed instructions. While highly effective, this approach lacks the capacity to learn from experience or adapt its guidance to novel environments or the specific preferences of an individual user, highlighting a key limitation of non-adaptive systems.

To address the rigidity of rule-based logic, recent research has pivoted towards adaptive decision-making using Reinforcement Learning (RL). The work by [58] exemplifies this cutting-edge approach by training an autonomous robotic guide dog to navigate complex environments. In this framework, the navigation agent learns an optimal policy by mapping its visual inputs directly to movement commands through continuous interaction with its surroundings, guided by a reward signal that encourages safe and efficient goal completion. This represents a significant leap towards truly intelligent and personalized assistance, as the agent can develop more sophisticated and flexible navigation strategies than could be explicitly programmed by a human. However, this approach also introduces the primary challenge of conventional RL: a reliance on trial-and-error learning, which is inherently unsuitable

for applications where human safety is paramount. The system's focus on goal achievement, with safety treated as a component of the reward function rather than a strict constraint, underscores the critical need for a more robust safety framework in adaptive navigation aids.

Further exploring the domain of real-time computer vision for assistive navigation, the system proposed by [10] demonstrates a foundational approach centered on immediate environmental perception. Their framework utilizes an RGB-D camera to capture both color and depth information, which is processed to detect floor planes, identify potential obstacles, and calculate safe walking paths. The guidance is delivered to the user through auditory feedback, translating complex visual data into intuitive directional commands. This work underscores the importance of real-time processing and reliable depth perception as core components of any ETA. However, similar to other systems of its era, its logic is inherently reactive and non-adaptive. The system navigates based on a fixed set of rules for obstacle avoidance and does not possess the capability to learn from user interactions or adapt to the nuances of previously unseen environments, highlighting the need for more intelligent, learning-based navigation strategies.

Addressing the limitations of rule-based systems, [48] introduce a navigation paradigm based on Reinforcement Learning (RL), specifically using a Deep Q-Network (DQN) algorithm. Their work shifts the focus from simple obstacle avoidance to goal-oriented, adaptive pathfinding within a simulated environment. The RL agent learns to navigate by taking actions (e.g., moving forward, turning) based on its current state and receiving rewards or penalties, allowing it to develop an optimal policy for reaching a destination safely and efficiently. This approach represents a significant advancement towards creating ETAs that can learn and adapt. The primary limitation, however, is its implementation within a simulation. While this is a necessary first step for safely training an RL agent, it raises critical questions about the model's ability to transfer its learned knowledge to the complexities and unpredictability of the real world—a challenge known as the "sim-to-real" gap.

Recognizing that sophisticated perception and navigation logic must be paired with an effective user interface, the work by [43] focuses on the crucial role of the feedback system. Their research investigates and develops a multimodal feedback interface that combines auditory cues with haptic (vibrational) feedback to convey detailed environmental information. The system uses computer vision to detect obstacles and landmarks, then translates this information into a rich set of signals, such as directional vibrations in a wearable belt and spatialized audio alerts. By engaging multiple sensory channels, this approach aims to reduce the cognitive load on the user and provide a more intuitive and immersive understanding of their surroundings. This research highlights that the ultimate success of an advanced navigation aid depends not only on the intelligence of its algorithms but also on its ability to communicate its insights to the user in a clear, timely, and non-overwhelming manner.

The application of machine learning to assistive technologies has matured into a diverse and impactful field, as surveyed in a recent review by [53]. This work provides a broad overview of the current landscape, mapping out the various ways ML is being deployed to enhance accessibility for the visually impaired. The author

highlights several key domains, including real-time object recognition for environmental awareness, advanced text-to-speech for accessing written information, and intelligent navigation systems for independent mobility. This comprehensive analysis establishes that machine learning is no longer a peripheral technology but a central pillar in the development of modern ETAs, providing a suite of powerful tools to address long-standing accessibility challenges.

Exemplifying the practical application of these advanced tools, [54] present a "Smart Assistive Stick" that leverages deep learning-based computer vision. Their device integrates a camera to capture the user's immediate environment and employs a sophisticated detection model to identify specific hazards and points of interest, including obstacles, potholes, and crosswalks. This perceptual information is then relayed to the user through a combination of vibrational and audio feedback. This research serves as a clear demonstration of how abstract deep learning capabilities are being successfully engineered into tangible, user-centric devices that significantly improve real-time safety and situational awareness during ambulation.

To further enhance the reliability of such systems, research has also focused on the principle of sensor fusion. The navigation aid developed by [42] showcases this approach by integrating data from multiple sources. Their system combines a camera, for rich visual perception processed by a YOLO object detection model, with an ultrasonic sensor for robust proximity detection. This multimodal design creates a more resilient and dependable system, as the ultrasonic sensor can effectively compensate for the inherent limitations of computer vision in challenging conditions like poor lighting or when faced with textureless surfaces. This work highlights a critical trend towards building ETAs that are not only intelligent in their perception but also robust and reliable across the wide spectrum of real-world environments.

Assistive devices for visually impaired people can be often expensive and uncomfortable to use. To address this, research paper[37] proposes DeepNAVI, a smartphone-based portable navigation assistant which uses deep learning to provide detailed information about position, type, distance and motion status of an obstacle. The EfficientDet-Lite4 model, a convolutional neural network (CNN) pre-trained with ImageNet database for classification, detects objects in real-time, estimates the position and recognizes the motion. Additionally, it recognizes and classifies the environment to 20 common scene types for indoor and outdoor environments. The instruction output is delivered through the audio. To enhance offline connectivity, the model collects dataset from the user by capturing images from the user environment. This can lead to privacy concerns regarding the potential collection of sensitive data. The paper [37] also fails to provide a solution to a potential data breach where the device is lost or stolen. Using smartphone cameras as input and EfficientDet-Lite4 as detection models may have limited functionalities in low-light environments.

A relevant topic of assistive technology for the navigation of visually impaired individuals is this research paper [50], which employs a custom safe reinforcement learning framework for mobile robot navigation in dynamically dynamic environments, ensuring both safety and efficiency. The agent is trained in an environment featuring both static and dynamic obstacles. The safety constraints set in this framework stops the agent from taking unsafe action. The agent continuously adapts its

path predicting the movement of the obstacles to avoid collisions. The navigation policy ensures the robot seeks the most optimal path involving minimal risk of collusion with the obstacles. The main goal of the paper [50] was to remove the conflict of exploration and exploitation. However, the model might not be functional in a complex environment where there are safety concerns other than avoiding obstacles. When it involves navigating for blind individuals, the average speed must also be faster.

The research paper [12] explores the idea of an affordable and efficient navigation aid for visually impaired people. The system focuses on detecting static and dynamic objects in real-time using a RGB camera and depth sensor of Microsoft Kinect capture. The depth image is processed and divided into vertical lines, for which a Distance Along Line Profile (DAP) graph is generated. The processed images then undergo further analysis such as movement detection and edge detection: which categorizes the scene into different classes. While the system is effective in an indoor environment, the Kinect sensor struggles in a setting where the unfavorable lighting [8] of an outdoor environment. Furthermore, the detection range of the system is limited to 800mm – 600mm, which is insufficient for outdoor urban areas. Moreover, the system does not have a real-time feedback mechanism for the visually impaired users.

This paper[55] proposes a mobile application using machine learning algorithms and computer vision techniques to implement text recognition, object detection and currency detection. It introduces Reward Sequence Distribution conditioned on the starting observation and predefined action sequence(RSD-OA) to capture task-relevant information without visual distractions, in which the existing Visual Reinforcement Learning(VRL) shows degraded performance. The proposed CRESP significantly improves generalization performance in unknown environments and outperforms the state-of-the-art methods. This system is designed to solve the problem of accessibility of the technology for the market and adds haptic feedback for object detection if the user is too close. It also provides an added feature of currency detection which removes the problem for recognizing old, changed or foreign currency. It is a paper made based on demand on the market analysis of such assistive technology. The proposed system seems to give limited attention to dynamic terrains, traffic lights, traffic signs and moving vehicles and pedestrians.

In this paper[32], the proposed system uses a robotic walker(ISR-AIWALKER) that was validated in both virtual and real environments to help in navigation for users with gait disorders. The paper developed a two-stage approach that combines reinforcement learning(RL) with rapidly-exploring random tree(RRT) algorithms to enhance the safety and efficiency of navigation based on object detection. It was made to find safe terrains based on the user's intent providing full control of the movement until obstacles are detected. In dangerous scenarios where obstacles are detected, corrections are computed to avoid collisions. This system needs to adapt to varying needs and capabilities of users to ensure a personalized and effective navigation experience. The two-stage approach lacks real-time adaptability in changing environments where obstacles may appear suddenly. The real-time processing is crucial in this system which possesses the issue of catastrophic failure in confusing lighting conditions. The robot walker approach is power consuming which limits the

distance the user can travel.

According to the paper[26], Reinforcement Learning(RL) for safety strategies in Ambient Assisted Living(AAL) applications enhance the quality of life for cognitive impaired and Alzheimeirs individuals through intelligent support and monitoring. This approach defines a recovery process from dangerous situations through real-time Q-learning that ensures the AAL system can adapt to unpredictable and irrational behavior of the patients. It's a prototype for an intelligent agent that can recognize dangerous situations and take recovery steps like turning off a heater and sound alarm. The system has been tested in a simulated environment with various dangerous situations and the agent responds effectively by continuously learning and updating its strategies real-time reducing the need for pre-defined, static safety measures. The system was trained for specific situations and possesses the concern of information security issues. It seems to have the possibility of failure for large scale processing in complex and dynamically changing larger environments.

The research paper by Kirillov et al. [28] introduces Panoptic Segmentation (PS), a novel computer vision task that unifies semantic and instance segmentation. This approach assigns each pixel in an image both a class label (e.g., "road," "car") and an instance ID, creating a more comprehensive and coherent understanding of the scene. The paper argues that by treating "stuff" (amorphous regions) and "things" (countable objects) in a unified manner, PS overcomes the limitations of previous methods where semantic and instance segmentation were treated as separate tasks, which could lead to contradictory outputs. To evaluate this new task, the authors propose the Panoptic Quality (PQ) metric. The primary application discussed is autonomous driving, where a complete and detailed understanding of the environment is crucial for safe navigation. While the paper establishes a strong foundation for this unified task, it primarily focuses on introducing the concept and the evaluation metric, with less emphasis on the limitations of specific models implementing this approach.

The research paper by Cheng et al. [31] presents Panoptic-DeepLab, a single-shot, bottom-up system for panoptic segmentation designed to be simple, strong, and fast. The system's bottom-up architecture first assigns a semantic label to every pixel and then groups pixels belonging to "thing" categories into distinct instances. This is achieved by simultaneously predicting instance centers and regressing the offset from each foreground pixel to its corresponding center, which allows for a simple grouping operation. A significant impact of this efficient, single-pass design is its suitability for edge computing, as it can achieve near real-time inference speeds (15.8 frames per second) when paired with a lightweight backbone like MobileNetV3. This provides a strong speed-accuracy trade-off for applications on resource-constrained devices. However, achieving the highest accuracy requires heavier backbone networks, which substantially reduces the inference speed, making it less viable for real-time edge applications. Additionally, the paper notes that the bottom-up approach can struggle with large objects, sometimes incorrectly segmenting them into multiple smaller instances due to large scale variation within the scene.

This paper dives into the use of dilated convolutions for multi-scale context aggregation in semantic segmentation [11]. Unlike traditional classification networks that rely on pooling and downsampling, the authors propose a context module that

expands the receptive field exponentially without reducing resolution. By stacking layers with increasing dilation rates, the network captures global context while preserving spatial detail—crucial for dense prediction tasks. The module is plug-and-play, compatible with existing architectures, and significantly boosts performance when added to simplified front-end models. The authors also re-examine adaptations of classification networks for segmentation, showing that removing unnecessary pooling layers and using dilated convolutions improves accuracy. Experiments on Pascal VOC 2012 demonstrate that their simplified front end, combined with the context module, outperforms prior models like FCN-8s and DeepLab, even without structured prediction. This work is foundational for your CARLA RL+CV pipeline, as it justifies using dilated convolutions to maintain resolution while expanding context—key for detecting dynamic obstacles in urban navigation.

This paper dives into semantic video segmentation using a Spatio-Temporal Transformer Gated Recurrent Unit (STGRU) that combines optical flow warping with adaptive gating mechanisms to propagate labeling information across video frames [21]. The authors address the computational challenge of processing large video datasets by augmenting static CNN architectures with recurrent layers that temporally fuse semantic estimates from unlabeled frames. Their methodology employs a modified GRU where the reset gate incorporates flow confidence measures computed by comparing consecutive warped images, allowing the network to adaptively weight temporal propagation based on motion certainty. Tested on CityScapes and CamVid datasets, the GRFP (Gated Recurrent Flow Propagation) model achieved 69.4% mIoU by leveraging five temporal frames, demonstrating 0.7-1.0 percentage point improvements over static baselines while maintaining temporal consistency with 85% label stability across trajectories. The end-to-end trainable architecture enables joint optimization of recognition, optical flow, and temporal propagation modules, though computational costs remain significant at 335ms per frame with high-quality optical flow estimation.

This paper explains a lightweight CNN-GRU architecture for real-time road segmentation in autonomous vehicles, achieving 86.91% F1-score on KITTI benchmarks while operating at 50 frames per second [25]. Unlike traditional encoder-decoder networks requiring extensive computation, the authors frame road segmentation as a semantic captioning task where GRU networks process spatial sequences of CNN-extracted features to predict road boundaries. The architecture introduces coordinate channels alongside RGB inputs to exploit spatial coherence in road positioning, implements shallow 11×11 convolution kernels followed by 1×1 layers for efficient feature extraction, and employs bidirectional GRUs with 256 hidden neurons to capture left-right context. A pyramid prediction scheme handles scale variance between near and far road regions by separately processing cropped and scaled image frames. With only 2.79 million parameters and 1.97 gigaFLOPs—dramatically fewer than StixelNet's 6.82M parameters and 43.0G FLOPs—the model achieves comparable accuracy with 91.24% precision and 4.39% false positive rate, demonstrating the efficiency advantages of recurrent architectures over fully convolutional approaches for structured perception tasks in resource-constrained embedded systems.

This paper explains a Geo-ConvGRU framework that addresses temporal dependency limitations in Bird's-Eye View (BEV) segmentation by replacing 3D CNN

layers with Convolutional Gated Recurrent Units augmented with geographical masking [57]. The authors demonstrate empirically that 3D CNNs exhibit bounded performance improvements as temporal receptive fields expand beyond 4–5 frames, while ConvGRU achieves consistent accuracy gains with extended sequences at significantly lower computational cost compared to Transformer-based alternatives like BEVFormer. The core innovation involves substituting fully connected layers in standard GRUs with convolutional operations to process spatial features, formulated as: $z_t = \sigma(W_z * f_t + U_z * h_{t-1})$ for the update gate, $r_t = \sigma(W_r * f_t + U_r * h_{t-1})$ for the reset gate, and $h_t = (1 - z_t)h_{t-1} + z_t\tilde{h}_t$ for the hidden state update, where $*$ denotes convolution. Additionally, a geographical mask M_{geo} derived from camera intrinsic/extrinsic parameters suppresses invalid voxels—those without corresponding 2D pixel projections during BEV transformation—thereby mitigating spatial noise introduced during temporal fusion that causes false predictions for moving objects. Evaluated on NuScenes dataset across three tasks (semantic segmentation, instance segmentation, and map prediction), Geo-ConvGRU achieves 1.3%, 0.9%, and 0.8% improvements over state-of-the-art methods respectively, while maintaining 6.4 Hz inference speed with only 19.8-hour training time compared to BEVFormer's 64.3 hours and 1.9 Hz, demonstrating superior efficiency-accuracy tradeoffs for real-time autonomous driving applications.

This paper elaborates on safe autonomous navigation in CARLA using Deep Q-Networks with semantic segmentation and depth cameras for obstacle detection. The study demonstrates collision-free navigation by preprocessing sensor outputs to reduce state space complexity before feeding into the RL model. The hierarchical architecture separates braking decisions from steering control, achieving high success rates across multiple trajectories. Key relevant contributions include the sensor fusion approach combining segmentation with depth information, and the two-stage decision model (braking vs. driving) that mirrors safety-aware control. However, shortcomings include simulation-only validation, limited algorithm comparison beyond DQN, and lack of temporal prediction for dynamic obstacle trajectories [46].

This paper dives into systematic analysis of reinforcement learning approaches for autonomous driving in CARLA, revealing that model-free methods like PPO dominate the field (80%+ of studies). The review critically examines safety limitations in current RL approaches, identifying that most methods lack formal safety guarantees and struggle with constraint satisfaction during exploration. Key relevant findings include: PPO's widespread adoption for continuous control, the challenge of designing effective reward functions that balance task performance with safety, and the prevalence of vision-based observation spaces using semantic segmentation. The survey identifies action shielding and constrained RL as emerging solutions for safety-critical navigation. Major shortcomings across reviewed literature include inadequate temporal reasoning for dynamic obstacles, poor sim-to-real transfer, and absence of formal safety verification mechanisms during training [59].

This paper talks about formalizing safe exploration through Constrained Markov Decision Processes (CMDPs), where safety requirements are specified separately from task objectives using auxiliary cost functions. The authors propose action shielding mechanisms that evaluate action safety before execution, filtering unsafe actions through hard constraints—conceptually similar to action-specific pixel safety

evaluation. They benchmark PPO with Lagrangian constraint enforcement, demonstrating that adaptive penalty methods more reliably satisfy safety constraints than direct constrained optimization (CPO). The Safety Gym environments separate reward functions (task performance) from cost functions (safety violations), enabling explicit measurement of constraint satisfaction during training. However, limitations include simulation-only evaluation, challenges with complex locomotion under constraints, and lack of temporal prediction for anticipatory safety, which your GRU-based approach addresses [29].

This paper dives into a formal mathematical framework for reinforcement learning problems where safety is a primary concern. It introduces efficient policy gradient and actor-critic algorithms designed for risk-constrained environments, representing risk with metrics like Conditional Value-at-Risk (CVaR). For our project, this provides a principled way to implement our "PPO with safety layer" by defining collision-free navigation as a formal probabilistic constraint that the algorithms can directly optimize. However, a notable limitation is that the algorithms are validated on less complex problems than CARLA's dynamic environment, and the convergence proofs only guarantee finding locally optimal policies, which may not be the absolute best solution for your agent [24].

This paper elaborates on an algorithm named Constrained Proximal Policy Optimization (CPPO). It tackles the exact challenge of enforcing safety constraints within PPO efficiently by using a novel, first-order method that avoids the complexity of traditional techniques. The approach reframes the problem into a two-step process: first calculating a safe policy, then updating the current policy towards that target. While this offers a streamlined way to train an agent, it is important to note that the paper is a preprint that has not undergone full peer review, and its effectiveness is demonstrated in standard benchmark environments that may not fully capture the unique challenges of vision-only navigation in a complex simulator like CARLA [49].

This paper introduces algorithms for Robust Constrained Markov Decision Processes (RCMDPs), which can help address the "sim-to-real" gap. It talks about ensuring a policy is safe not just in its training environment but also robust to uncertainties and changes in the real world. The core idea is to train the policy against an "adversary" that finds the worst-case version of the environment, making the agent more resilient. While this could enhance your agent's robustness, the proposed adversarial training approach is computationally intensive, which may conflict with our goal of a lightweight architecture. Furthermore, it introduces the added complexity of training a second, adversarial policy, which can be difficult to tune and stabilize during training [52].

This paper introduces a formal method for ensuring safety in reinforcement learning by synthesizing a "shield" from a temporal logic specification. The shield operates independently of the learning agent, either by preemptively offering only safe actions or by monitoring and correcting the agent's choices if they violate the specification. This creates a hard separation between safety enforcement and reward optimization. While this provides strong safety guarantees, its primary shortcoming is its reliance on a pre-existing, accurate abstraction of the environment's dynamics. For complex,

high-dimensional, or unknown environments, creating such a faithful model upfront is often impractical or computationally prohibitive, which significantly limits the method's applicability in real-world scenarios where system models are not readily available [15].

This work proposes "dynamic shielding" as an extension to handle black-box environments where a system model is not available beforehand. The approach learns an approximate system model online, in parallel with the reinforcement learning agent, and uses this evolving model to construct and update the safety shield. This allows the agent to foresee and prevent unsafe actions based on its growing experience. However, the method's effectiveness is constrained by the quality of the learned model. Early in training, the model is sparse and inaccurate, which can cause the shield to be overly conservative, preventing necessary exploration and sometimes leading to longer training times compared to unshielded methods. The process also introduces a non-trivial computational overhead for continuously learning the model and updating the shield [38].

This paper introduces the Xception architecture, a powerful and efficient convolutional neural network model built upon the principle of depthwise separable convolutions. The work is founded on the insightful hypothesis that cross-channel and spatial correlations in feature maps can be effectively decoupled. This allows standard convolutions to be replaced by a more efficient sequence of a depthwise convolution (which acts on each channel independently) and a pointwise convolution (which combines the outputs). This design choice dramatically reduces computational cost and the number of model parameters without sacrificing performance. The paper demonstrates the strength of this architecture, showing that Xception outperforms the highly influential Inception V3 architecture on large-scale image classification tasks, thereby establishing depthwise separable convolutions as a cornerstone of modern, efficient deep learning models [17].

This paper presents the highly influential DeepLab system, which makes significant strides in semantic image segmentation. Its key contribution is the introduction of (ASPP), an innovative module that addresses the critical challenge of segmenting objects at multiple scales. ASPP utilizes parallel atrous (dilated) convolutions with varying rates to capture rich, multi-scale contextual information from feature maps efficiently. This allows the network to robustly understand both fine-grained details and the broader scene context. The paper also successfully integrates a fully connected Conditional Random Field (CRF) as a post-processing step. This hybrid approach elegantly combines the rich feature hierarchies of deep neural networks with the fine-grained localization accuracy of probabilistic graphical models, leading to sharp and accurate object boundaries [16].

This paper proposes ECA-Net, a novel and exceptionally lightweight Efficient Channel Attention module that significantly boosts the performance of deep convolutional neural networks with negligible computational overhead. The work cleverly improves upon prior channel attention mechanisms by demonstrating that avoiding dimensionality reduction and employing a local cross-channel interaction strategy is highly effective. This is ingeniously implemented using a fast 1D convolution whose kernel size is adaptively determined based on the channel dimension. ECA-Net stands out

for its simplicity and efficiency, proving that substantial performance gains can be achieved without the complexity and computational burden of more sophisticated attention mechanisms. It provides a flexible, plug-and-play module that enhances model performance across a variety of vision tasks [33].

2.3 Summary of Key Findings

From the above discussion, it is observed that most of the existing research focuses on assistive technologies for the visually impaired using similar approaches as in designing navigation systems for different purposes. However, one must consider the unique challenges faced by visually impaired pedestrians for reading road markings, traffic signals, and moving objects in a dynamically changing environment with changing light. Traditional assisted navigation aids that depend on pre-installed infrastructure or lack real-time adaptability are not well-suited for these dynamic settings. Furthermore, navigational areas may be complex with both static and dynamic obstacles, they require solutions that can provide real-time feedback and adapt to constantly changing surroundings. In addition, the existing models have the risk of user data breach while accessing personal data for personalization based on user and environment. Therefore, the approaches to assistive navigation should not be limited to static object detection but should extend to real-time, adaptive systems capable of handling dynamic urban elements, such as moving vehicles and pedestrians, to ensure safety and mobility for the visually impaired along with protected user data.

The purpose of this assistive navigation model is to ensure safe optimized navigation for visually impaired pedestrians in the urban environment. In order to achieve that the model needs to take images of the environment as input. The navigation instruction received by audio or haptic feedback would be the output. The entire model can be broken down into two separate part: Image processing and Decision making. 4.1 provides a simplified view of the model design.

Chapter 3

Requirements, Impacts and Constraints

3.1 Final Specifications and Requirements

The primary objective of this project is to develop a safe navigation system for a visually impaired pedestrian agent in urban environments using vision-only reinforcement learning. The functional requirements were defined to prioritize collision avoidance and goal-oriented movement in dynamic scenarios. The system must process RGB camera inputs (512×1024 resolution, 90° FOV) mounted on the agent's head, perform real-time semantic segmentation using MobileNet-based + FPN 19 object classes trained on Cityscapes, and predict future obstacle positions using a ConvGRU network with 5–10 frame temporal windows. The RL agent, implemented with Proximal Policy Optimization (PPO) through Stable Baselines3, must support four discrete actions (stop, forward, left, right) and incorporate a safety layer that filters unsafe actions by analyzing action-specific pixel regions from the augmented semantic mask. Technical specifications include CARLA simulator version 0.9.16 or higher, Python 3.12+, PyTorch for deep learning components, and OpenCV, PIL for image processing. The system must achieve a success rate exceeding 80% in reaching navigation goals while maintaining collision rates below 10%, with inference running at minimum 10 FPS to ensure real-time responsiveness.

Non-functional requirements emphasize robustness, safety, and computational efficiency to align with real-world assistive technology standards. The system must comply with ISO 9241-171 guidelines for accessibility software and follow ethical AI principles outlined in IEEE's Ethically Aligned Design framework, particularly regarding autonomous systems that assist vulnerable populations. Performance constraints include maximum latency of 50ms per decision cycle, memory usage under 4GB to enable deployment on edge devices, and stable operation across varied weather conditions and urban layouts in CARLA's Town01–Town03 environments. The ConvGRU prediction module must be pre-trained on 5,000–20,000 supervised trajectory sequences using both DICE and Focal Loss before integration into the end-to-end RL pipeline. Safety constraints mandate that the post-policy safety layer must override any action predicted to result in collision based on semantic obstacle detection thresholds (> 0.1 for static obstacles, > 0.5 for predicted danger zones). All training procedures must ensure reproducibility through fixed random seeds, and the system architecture must support future extension to monocular

depth estimation for fully vision-based distance computation, moving toward eliminating dependence on CARLA's API position data. For future implementation of a device, surround sound cues along with haptic feedback will be used to guide the user of their surroundings.

3.2 Societal Impact

This project addresses critical societal challenges in mobility and independence for the 2.2 billion people worldwide with visual impairments, as reported by WHO. By developing vision-only safe navigation technology, the system has the potential to significantly enhance quality of life by enabling autonomous urban travel without constant human assistance, reducing social isolation and promoting workplace inclusion. From a health and safety perspective, the collision avoidance mechanisms directly protect users from physical harm in dynamic traffic scenarios, while the predictive danger detection reduces cognitive load and navigation-related anxiety. However, safety concerns exist regarding system reliability—false negatives in obstacle detection could lead to injuries, necessitating rigorous testing before real-world deployment and clear user warnings about system limitations.

3.3 Environmental Impact

Implementing a vision-only safe RL pedestrian in CARLA has modest environmental impact when developed responsibly. Simulation-based training reduces need for physical trials, lowering emissions and resource use compared with real-world data collection. Compute energy from model training (segmentation, GRU, PPO) is the primary footprint; we mitigate it via efficient architectures (MobileNet, compact GRU), mixed-precision training, and reuse of simulated datasets. Transfer experiments should minimize physical prototyping and follow proper disposal of hardware. Overall, simulation-first design offers sustainable development benefits, but researchers must monitor and report compute energy, optimize training, prefer carbon-aware eco-friendly resources, and prioritize renewable energy sources whenever possible.

3.4 Ethical Issues

Designing a vision-only safe-RL pedestrian entails ethical concerns and professional duties. Safety and reliability are paramount: developers must prevent harm by rigorous testing, conservative deployment, and human oversight. Privacy arises if real-world data are collected—ensure consent, anonymization, and lawful handling. Bias and fairness require diverse scenarios to avoid unsafe behavior for particular groups. Transparency, explainability, and clear documentation of limitations, failure modes, and datasets are professional obligations. Researchers must avoid overstating capabilities, disclose environmental costs, and follow legal/regulatory standards. Finally, plan for responsible reuse and mitigate misuse by sharing code, data, and safeguards under appropriate licenses, and engage diverse stakeholders.

3.5 Standards

This project adheres to three critical standards to ensure safety, accessibility, and ethical deployment of the vision-based navigation system. First, ISO 9241-171:2008 (Ergonomics of human-system interaction – Guidance on software accessibility) [7] establishes the foundational requirements for assistive technology design, ensuring the system interface and functionality accommodate users with visual impairments through appropriate feedback mechanisms and fail-safe protocols. Second, the Federal Highway Administration's (FHWA) Safety Surrogate Assessment Model (SSAM) [6] provides formal acceptance criteria through established conflict thresholds: Time-to-Collision (TTC) of less than 1.5–2.0 seconds and Post-Encroachment Time (PET) of less than 1.5 seconds, which our system must maintain below the violation rate per kilometer including the upper 95% confidence interval. Third, IEEE P7001 Standard for Transparency of Autonomous Systems [34] guides the implementation of explainable AI decision-making in our PPO safety layer, ensuring that action-filtering logic is auditable and interpretable for safety validation. These standards collectively frame the technical requirements, safety benchmarks, and ethical considerations that govern our development methodology and establish measurable compliance criteria for real-world deployment readiness.

3.6 Project Management Plan

A two-month project schedule divides tasks into phases: setup and data collection (1 week), segmentation model training (2 weeks), ConvGRU prediction development (1 week), RL integration and safety-layer implementation (2 weeks), simulation experiments and ablation studies (1 week), and evaluation and documentation (1 week). Team: one lead researcher, one research engineer, one data annotator, and two advisors. Management: weekly sprints, biweekly reviews, version control, a risk register, and milestone-based deliverables to ensure timely progress. Regular stakeholder demos and budget reviews are planned.

3.7 Risk Management

Risk management in this project prioritizes safety, transparency, and system reliability throughout both the simulation and model training lifecycle. Since the reinforcement learning (RL) agent is trained exclusively within the CARLA simulator [19], all exploration and policy adaptation occur in a controlled, risk-free virtual environment, eliminating the possibility of real-world harm. This aligns with the ISO 9241-171:2008 standard on accessibility and ergonomic interaction, ensuring that all algorithmic decisions are validated through safe, user-centric feedback channels before deployment. Additionally, adherence to the FHWA Safety Surrogate Assessment Model (SSAM) thresholds—Time-to-Collision (TTC ¡ 1.5–2.0 s) and Post-Encroachment Time (PET ¡ 1.5 s)—guides our collision-avoidance evaluation and risk quantification.

Primary technical risks include unsafe policy exploration, poor generalization from simulated to unseen environments, ConvGRU prediction failure under novel dy-

namics, and cascading segmentation errors. To mitigate these, the Proximal Policy Optimization (PPO) agent [22] is equipped with a safety layer conforming to IEEE P7001 principles of transparent and interpretable decision logic. The layer enforces action shielding [23], which filters high-risk actions before execution, ensuring that the agent never violates safety constraints during learning or inference. Additional mitigations include (i) domain randomization across CARLA towns, weather, lighting, and traffic densities, (ii) cautious online fine-tuning using small learning rates, and (iii) ensemble averaging and temporal smoothing to stabilize predictions.

Operational risks—such as compute cost overruns, dataset bias, and inadequate incident logging—are mitigated through contingency budgeting, routine bias audits, and secure data governance frameworks. The entire training and validation pipeline remains simulation-only, and all deployment candidates are subject to offline safety verification and institutional ethical review prior to any real-world application. Collectively, these measures ensure that the Safe RL navigation framework operates within strict safety, accessibility, and transparency constraints, minimizing both technical and ethical risks.

3.8 Economic Analysis

This project prioritizes computational efficiency through strategic architectural choices that enhance accessibility and deployability compared to existing solutions. By employing lightweight MobileNet-based semantic segmentation instead of resource-intensive backbones like ResNet-101, combining with parameter-efficient PPO reinforcement learning and memory-lean GRU temporal modeling, the system achieves real-time performance on edge hardware platforms such as Qualcomm Snapdragon processors without requiring specialized GPUs. Furthermore, relying solely on RGB camera inputs rather than expensive LiDAR sensors significantly reduces hardware requirements while maintaining robust perception capabilities. These design decisions collectively enable deployment on consumer-grade mobile computing devices, democratizing access to advanced assistive navigation technology and facilitating broader adoption in resource-constrained environments where conventional autonomous systems remain impractical due to computational and hardware demands.

Chapter 4

Proposed Methodology

4.1 Design Process or Methodology Overview

The design process for this research is fundamentally guided by the primary objective: to develop a safe and reliable navigation system for visually-impaired users as a pedestrian that can operate effectively on a computationally constrained, battery-powered wearable device. This core constraint necessitates a dual-axis evaluation methodology where every architectural decision is judged simultaneously on its safety guarantees—both during the model's training phase and in real-world deployment—and its computational efficiency, measured by metrics such as end-to-end latency, frames per second (FPS), and power consumption. Consequently, our

methodology is structured around a comparative analysis of model families whose performance metrics are known to correlate with downstream safety outcomes. We selected Panoptic Quality (PQ) for its ability to provide a unified understanding of a scene, mean Intersection over Union (mIoU) for its precision in assessing free-space, and Average Precision (AP) for its accuracy in detecting individual agents. These perception metrics are paired with surrogate safety thresholds adopted from established traffic safety engineering practices, specifically Time-to-Collision (TTC) of less than 1.5–2.0 seconds and Post-Encroachment Time (PET) of less than 1.5 seconds, which serve as indicators of critical conflicts. These thresholds are not arbitrary; they are standard in the Federal Highway Administration's (FHWA) Safety Surrogate Assessment Model (SSAM) and are utilized as formal acceptance criteria for our final design. The chosen system must maintain a violation rate per kilometer, including its upper 95% confidence interval, below these established limits.

4.2 Preliminary Design

To identify the optimal architecture, we evaluated eight distinct perception–planning alternatives. The evaluation framework was comprehensive, assessing each alternative against a suite of metrics: perception accuracy (mAP, mIoU, PQ), task performance (goal-reach rate, path inefficiency), user comfort (jerk, hard-stops), and deployment viability (FPS, latency, power).

On the perception front, we compared several model families. While semantic-only segmentation models like DeepLabv3+ offer excellent free-space delineation

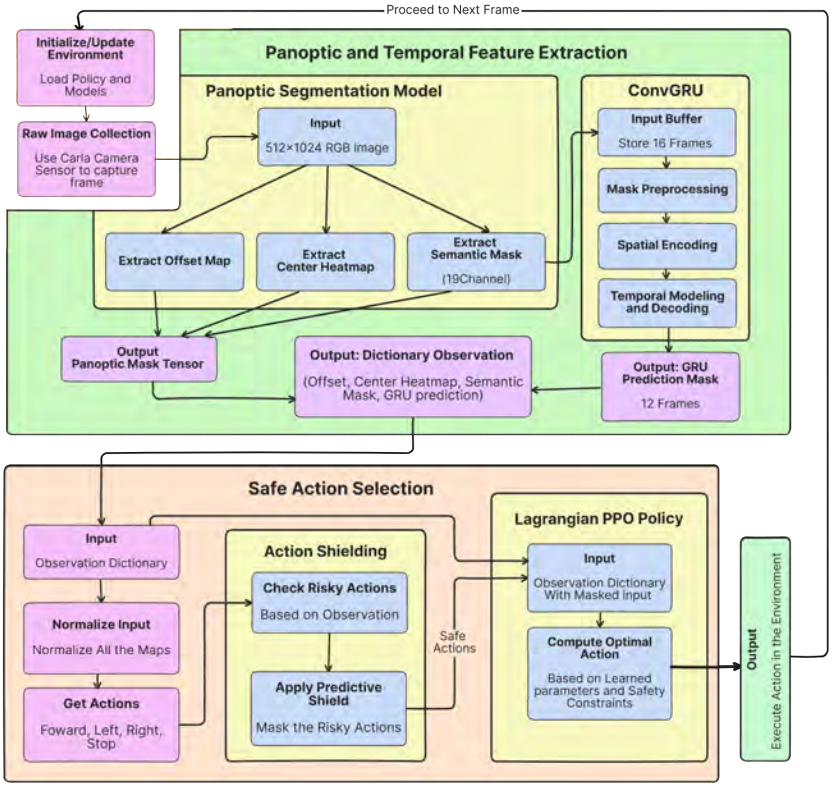


Figure 4.1: System Architecture.

(82.1% mIoU on Cityscapes), they lack the ability to distinguish individual agents in crowded scenes. Instance-only segmentation improves agent separation, but fails to specify background affordances crucial for alignment with curbs and sidewalks. Panoptic segmentation models, which unify "things" (dynamic agents) and "stuff" (static background), proved most compelling. We evaluated two primary panoptic variants: a lightweight Panoptic-DeepLab with a MobileNetV3 backbone (54.1 PQ @ 15.8 FPS) optimized for real-time performance on wearables, and a heavier variant with an Xception-71 backbone (65.5 PQ) offering superior accuracy at a higher computational cost.

A standard PPO baseline offers stable on-policy learning via a clipped surrogate, but it does not directly satisfy a cost limit, which motivated constrained alternatives for a safety-focused application. PPO-Lagrangian introduces an adaptive dual variable that balances reward and cost to meet a target budget on safety benchmarks, whereas CPO uses trust-region constrained updates aimed at tighter adherence to cost limits during learning. To further reduce residual violations, a runtime safety shield filtered unsafe actions at execution time, complementing CMDP training signals with deployment-time safeguards. Classical modular stacks and planners available in CARLA served as reproducible baselines for comparison with the RL controllers under identical scenarios and metrics.

Environment development began with a custom bird's-eye-view pipeline from monocular depth using MiDaS, whose robust cross-dataset relative depth is intentionally scale/shift ambiguous, complicating metric BEV construction and downstream planning. To reduce complexity while gaining realism, controllability, and repeatability,

training and evaluation were moved to CARLA, which offers configurable sensors, weather, traffic, and standardized urban driving tasks suited to both modular and RL-based stacks [19].

For trajectory reasoning, an initial GRU that consumed centroids and velocities extracted from semantic masks underperformed because it discarded pixel-level spatial context critical in cluttered scenes. A ConvGRU was adopted to model spatiotemporal features on 2D maps by convolutional gating, improving fidelity at the cost of higher computation during training. To retain accuracy while controlling cost, feature maps were downsampled before ConvGRU layers, yielding faster training while preserving essential spatial structure for prediction and control [51].

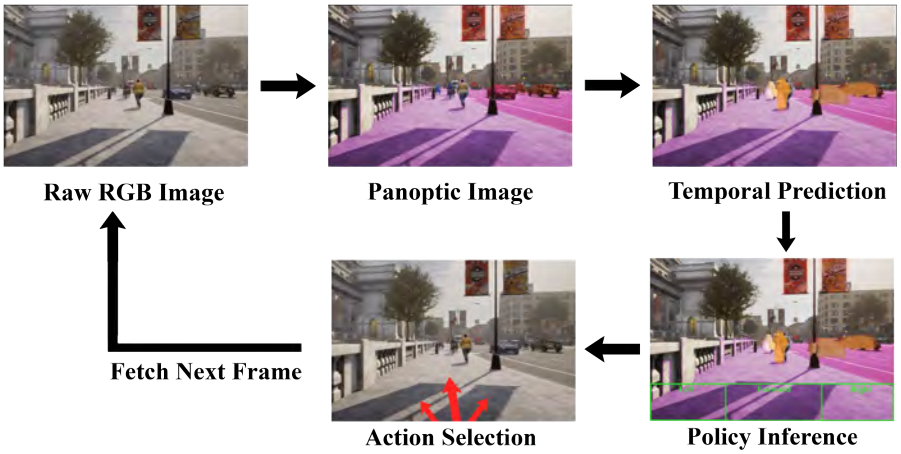


Figure 4.2: Process Visualization.

4.2.1 Simulation Results

Simulations were conducted separately, which was configured to stress-test failure modes relevant to pedestrian navigation, including crowded sidewalk navigation, crossings with variable driver compliance, and scenarios with occlusions. The results confirmed the superiority of the panoptic segmentation approach.

Table 4.1: Perception (Computer Vision) Baselines from Standard Datasets

Model (Family)	**Backbone**	**PQ**	**mIoU**	**Notes**
Panoptic-DeepLab (panoptic)	MobileNetV3	54.1	—	Near-real-time performance; good unified scene understanding at low compute.
Panoptic-DeepLab (panoptic)	Xception-71	65.5	84.2	Higher accuracy and stronger instance separation; heavier compute footprint.
DeepLabv3+ (semantic)	Xception	—	82.1	Excellent free-space segmentation; lacks per-instance reasoning for crowded scenarios.

The comparative analysis reveals a clear speed-accuracy frontier. The MobileNetV3-based panoptic model provides a viable, real-time solution for wearables, while the Xception-71 backbone delivers a higher safety margin in dense scenes at the cost

Table 4.2: Controllers (Reinforcement Learning) Used in Simulation

Controller	**Constraint Handling**	**Final Safety vs. PPO**	**Return / Path Efficiency**
PPO (Unconstrained)	None	Baseline (higher violations/km)	Often slightly higher when unconstrained
PPO-Lagrangian	Soft constraints via learned Lagrange multiplier	Better (lower violations/km at convergence)	Small, often negligible trade-off vs. PPO
CPO	Trust-region update with explicit constraint satisfaction	Better (near-constraint satisfaction)	Small-moderate trade-off; stricter, slower updates

of increased latency. On the control side, both PPO-Lagrangian and CPO demonstrated superior safety performance over unconstrained PPO by significantly reducing violation rates during training and at convergence. This implies that for a lightweight, battery-frugal wearable, the MobileNet-Panoptic perception core paired with a constrained controller like PPO-Lagrangian offers the most dependable safety margin within the given compute constraints.

4.3 Data Collection

Sources of data and methods of collection. To train and validate our models, we required a dataset that captured the diversity and edge cases inherent in real-world pedestrian navigation. We constructed our dataset by combining two publicly available, large-scale street-scene datasets: Cityscapes and Mapillary Vistas v2.0.

Cityscapes provides 5,000 finely annotated high-resolution (2048×1024) images from 50 European cities. Its consistent, car-mounted viewpoint and precise polygonal labels for 30 classes provide a high-quality baseline for training models to understand clean curb, crosswalk, and sidewalk geometry.

Mapillary Vistas v2.0 complements this with 25,000 globally crowd-sourced images captured from various devices (smartphones, dashcams) under diverse weather, seasons, and lighting conditions. Its dense annotations across 124 categories introduce the long-tail visual diversity needed to harden models against occlusions, clutter, and adverse conditions, thereby improving generalization.

This two-source approach was crucial. We used Cityscapes to establish baseline performance and tune core segmentation tasks, then introduced Mapillary Vistas v2.0 to expose the models to significant distribution shift, ultimately producing safer and more resilient navigation policies. Both datasets were used in accordance with their respective research licenses.

4.3.1 Data Cleaning

Our panoptic supervision pipeline encodes instance maps as 8-bit PNG images, which limits the number of distinct object instances in a single frame to 255. During preprocessing of the Mapillary Vistas v2.0 dataset, exactly five images identified that exceeded this limit. To prevent data corruption from instance ID collisions

Table 4.3: Public Datasets Used for Collection and Simulation

Dataset	**Scale & Resolution**	**Label Taxonomy & Annotation Style**	**Collection Method / Geography**	**Relevance to Project**
Cityscapes	5,000 fine + 20,000 coarse; 2048×1024	30 classes; pixel-accurate polygons; instance labels	Car-mounted stereo video; European cities	Clean curb/crosswalk geometry for cost-map construction and panoptic training.
Mapillary Vistas v2.0	25,000 high-resolution images	124 categories; dense polygon masks	Crowd-sourced street-level imagery; worldwide	Adds diversity (weather, lighting, clutter) to harden models and improve generalization.

or wrap-around, which would degrade label integrity and undermine safety-critical training, a conservative exclusion policy was adopted. These five images were omitted from both the training and evaluation sets, i.e., (`2Ix3yvNJY9fwQdZWum3t9g`, `FblzQaJ99XaqFUta4-_dKw`, `n6IjJovKemd8bIByGOew2Q`, `rcwrGs3OTinqod7_n0CImw`, `q8v6d4gPz2_nyAUwOkPLJw`). Their identifiers were recorded in a persistent skip list to ensure deterministic filtering and reproducibility across all experiments. This exclusion had a negligible impact on the overall data distribution while materially improving the fidelity of the labels.

4.3.2 Data Transformation

The data transformation pipeline was designed to standardize spatial resolution, improve robustness to environmental variations, and preserve the integrity of supervision labels. All images and masks were resized to a target resolution of 260×512 pixels.

- **Image Transformations:** For the training set, RGB images underwent a sequence of augmentations. After resizing, a random horizontal flip ($p = 0.5$) was applied to encourage left-right invariance. Color jittering (brightness, contrast, saturation, and hue) was used to simulate variations in daylight and camera properties. Finally, images were converted to tensors and normalized using standard ImageNet statistics. For validation and testing, only resizing and normalization were applied to ensure an unbiased evaluation.

- **Mask Transformations:** In contrast, supervision masks (both semantic and panoptic) were transformed using only a nearest-neighbor resizing operation. This method was chosen specifically because it preserves the discrete integer values of class and instance IDs, preventing the label mixing and boundary artifacts that would be introduced by other interpolation methods like bilinear or bicubic. No other augmentations were applied to the masks.

4.3.3 Data Integration

To enhance the model's situational awareness, traffic signal state information was integrated into the dataset. A lightweight YOLO object detector was applied to identify traffic lights within the images. For each detected traffic light instance, its

operational state (stop, wait, or go) was inferred and stored as a sidecar attribute, keyed by the image and instance ID. This process enriches the panoptic labels with critical, real-time context for safe crosswalk negotiation without altering the core 8-bit mask taxonomy. Cases with low confidence were flagged for human review, with a conservative "unknown" fallback to ensure safety.

4.3.4 Summary of Preprocessed Data

The preprocessing pipeline yielded two final dataset variants to balance semantic richness with system constraints:

1. **Variant A (Mapillary Vistas v2.0, 124 labels):** This variant retains the full 124-category taxonomy from Mapillary Vistas, maximizing the contextual diversity available to the perception model. It uses an 8-bit panoptic encoding, with the five high-instance frames excluded as noted in the data cleaning process.

2. **Variant B (Cityscapes, 19 labels):** This variant maps all labels to the standard 19 evaluation classes used in the Cityscapes benchmark. This provides a coarser but highly stable supervision set, with no risk of instance overflow under the 8-bit encoding scheme.

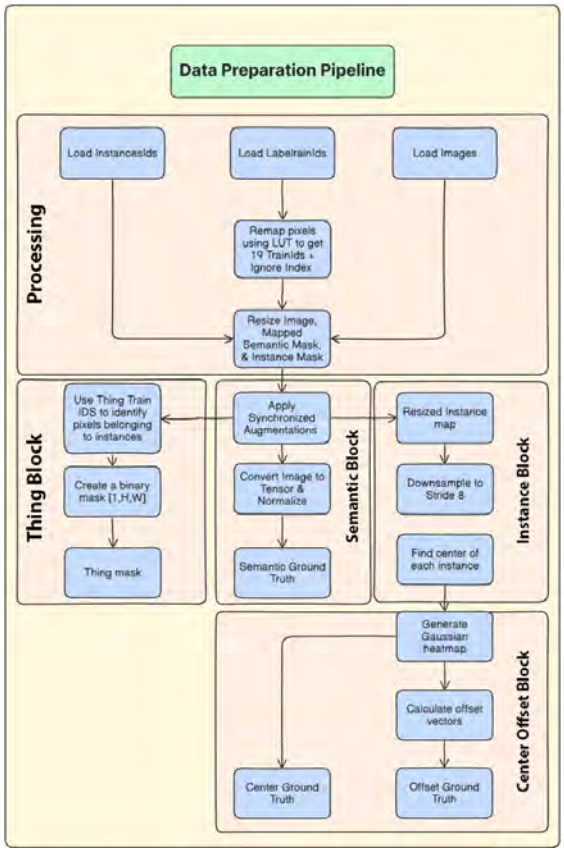


Figure 4.3: Data Preparation Pipeline.

In practice, the Vistas-124 variant was used to train models robust to visual diversity, while the Cityscapes-19 variant was used to ensure stable, overflow-free panoptic training. Both variants share the same 512×1024 resolution and asymmetric resampling policies.

4.4 Implementation of Selected Design

4.4.1 Initial Design and Baseline Implementation

Based on the preliminary design evaluation, we initially selected the following stack for implementation: a Panoptic-DeepLab (MobileNetV3) perception model, augmented with a short-horizon Convolutional GRU (ConvGRU) for motion prediction, and controlled by a PPO-Lagrangian reinforcement learning agent. This combination was chosen to meet the safety thresholds within the latency and power budget of a wearable device.
In this baseline implementation, the perception pipeline resized incoming RGB frames to 260×512 and normalized them. The Panoptic-DeepLab model processed these frames to produce a panoptic segmentation map. This map was then fed into the ConvGRU, which predicted a "future mask" at a short time horizon ($t + \Delta$) to better handle occlusions and near-term motion. From this output, a compact object list of the ten nearest "thing" instances was derived for the control module. The RL agent operated in a custom Gymnasium environment with a 50-dimensional state vector and a discrete action space (move-forward, turn-left, turn-right, stop). The PPO-Lagrangian algorithm was implemented using the Stable-Baselines3 library and trained to maximize a reward function that encourages smooth progress towards a goal while heavily penalizing collisions, curb encroachments, and unsafe proximity to other agents (TTC $<$ 2.0s, PET $<$ 1.5s).

4.4.2 Final Refined Architecture

While the initial configuration established the viability of the approach, further development led to a substantially refined and more deeply integrated architecture. The final design moves beyond a modular pipeline to a tightly-coupled system that optimizes the flow of information from perception to control, resulting in a more efficient and robust navigation agent. This architecture is a lightweight, single-network implementation that is notable for its proposal-free and Non-Maximum Suppression (NMS)-free instance recovery mechanism, making it exceptionally well-suited for real-time edge computing.

Perception Module

The core of the perception system is a jointly trained, end-to-end network designed for maximum efficiency and accuracy in panoptic segmentation. It processes a single RGB image and outputs dense predictions for both semantic segmentation (classifying every pixel) and instance geometry (identifying individual objects). These parallel predictions are later fused in a post-processing step to generate a unified panoptic output. The architecture, illustrated in Figure 4.4, consists of an efficient

Algorithm 1 Panoptic Segmentation Forward Pass

Require: Input image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ (see Fig. 4.4)
Ensure: Semantic segmentation map S, instance center heatmap H, pixel offset field O

Step 1: Multi-scale Feature Extraction
Extract backbone features at multiple scales:
    $C_2, C_3, C_4, C_5 \leftarrow \text{MobileNetV3}(\mathbf{I})$ ▷ Output strides: 4, 8, 16, 32

Step 2: Semantic Segmentation (Stuff Classes)
Enhance high-level features with contextual information:
    $C_5^{\text{ctx}} \leftarrow \text{ASPP}(C_5)$ ▷ Atrous Spatial Pyramid Pooling
Build feature pyramid by fusing multi-scale features:
    $P_2, P_3, P_4, P_5 \leftarrow \text{FPN}(C_2, C_3, C_4, C_5^{\text{ctx}})$
Generate semantic predictions at stride-4 resolution:
    $S_{\text{logits}} \leftarrow \text{SemanticHead}(P_2)$
    $S \leftarrow \text{Upsample}(S_{\text{logits}}, \text{factor} = 4)$ ▷ Full resolution
Generate auxiliary prediction for deep supervision (training only):
    $S_{\text{aux}} \leftarrow \text{Upsample}(\text{SemanticHead}(P_3), \text{factor} = 8)$

Step 3: Instance Segmentation (Thing Classes)
Apply lightweight context module on stride-16 features:
    $F_{16}^{\text{ctx}} \leftarrow \text{MiniContext}(\text{LRASPP-Lite}(C_4))$
Upsample context features to stride-8:
    $F_8^{\text{ctx}} \leftarrow \text{Upsample}(F_{16}^{\text{ctx}}, \text{factor} = 2)$
Project low-level features to match channel dimension:
    $F_8^{\text{low}} \leftarrow \text{ProjectorConv}(C_3)$
Fuse context and low-level features with channel attention:
    $F_{\text{fused}} \leftarrow \text{ECA}(\text{DSConv}(\text{Concat}(F_8^{\text{ctx}}, F_8^{\text{low}})))$
Predict instance centers as probability heatmap:
    $H \leftarrow \text{CenterHead}(F_{\text{fused}})$ ▷ Shape: $(H/8) \times (W/8) \times 1$
Predict pixel-to-center offset vectors:
    $O \leftarrow \text{OffsetHead}(F_{\text{fused}})$ ▷ Shape: $(H/8) \times (W/8) \times 2$

return Predictions: $\{S, S_{\text{aux}}, H, O\}$

backbone for feature extraction, a feature pyramid network for multi-scale fusion, and two specialized prediction heads for the semantic and instance tasks.

Backbone and Multi-Scale Feature Extraction: The foundation of the model is a MobileNetV3-Large backbone, pre-trained on the ImageNet dataset. This backbone is strategically chosen for its exceptional accuracy-to-latency trade-off, a critical requirement for deployment on resource-constrained edge devices. Its architectural efficiency stems from the extensive use of depthwise separable convolutions—which factorize standard convolutions into cheaper depthwise and pointwise operations—and the non-linear h-swish activation function.

From this backbone, a feature extractor exposes four intermediate feature maps, or "taps," at different scales, forming a feature pyramid. As detailed in Table 4.4, these taps provide a rich, multi-scale representation of the input image. Early, high-resolution taps like c2 (stride 8) and c3 (stride 16) preserve fine spatial details, which

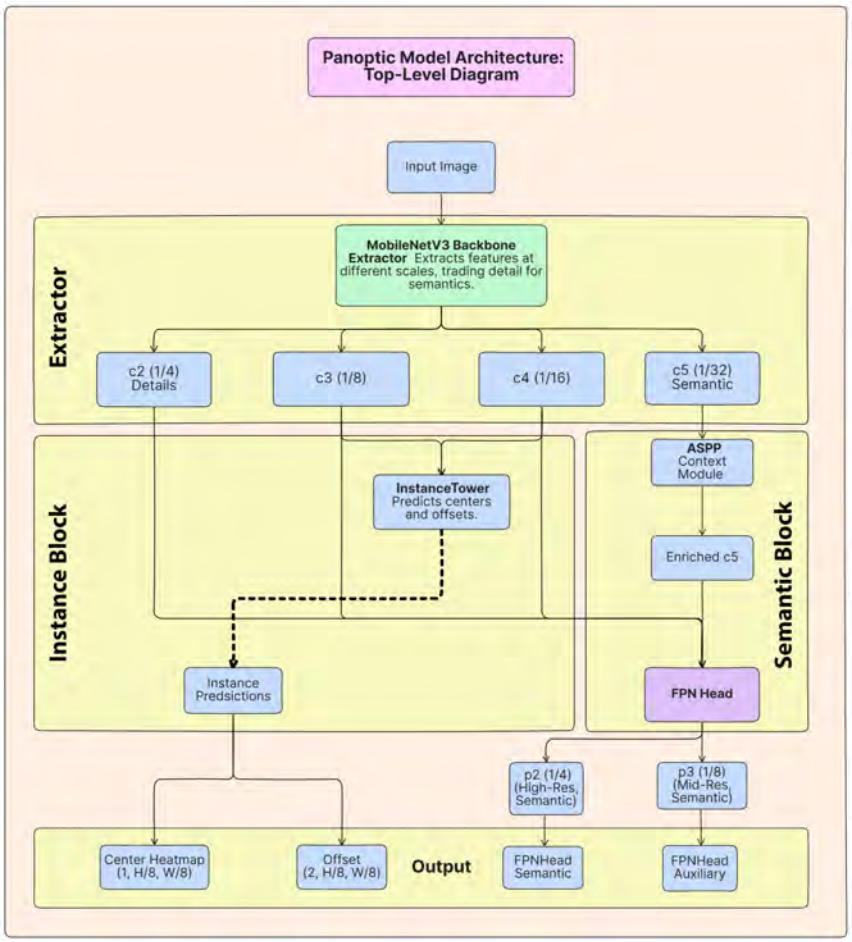


Figure 4.4: Panoptic Model Architecture.

are essential for precise instance localization and boundary delineation. In contrast, deeper, low-resolution taps like c4 and c5 (stride 32) have larger receptive fields and carry robust semantic information, which is vital for accurate class recognition.

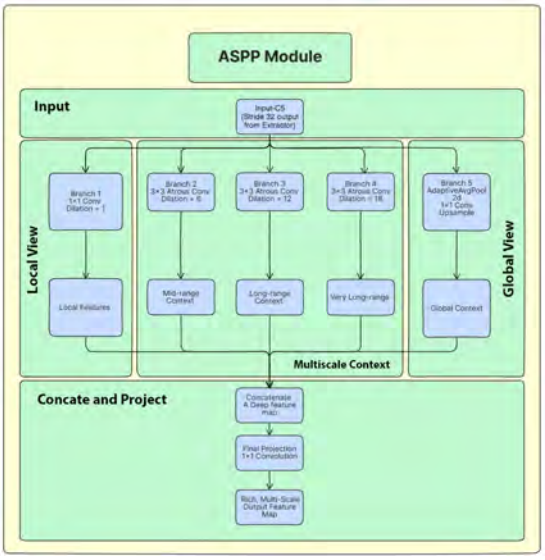
Table 4.4: Characteristics of Feature Maps at Different Levels.

Feature Map	Spatial Resolution	Semantic Information	Represents...
c2	High	Low	Edges, textures, colors
c3	Medium	Medium	Corners, simple shapes
c4	Low	High	Complex parts of objects
c5	Very Low	Very High	Entire objects or concepts

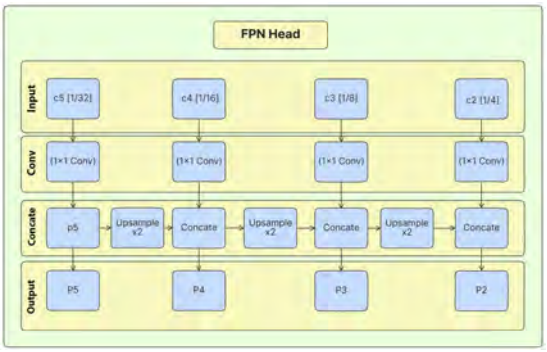
Semantic Segmentation Path: The semantic path is responsible for assigning a categorical class label (e.g., road, car, person) to every pixel in the image. This path is carefully designed to aggregate multi-scale contextual information for robust and accurate segmentation.

Context Aggregation with ASPP: To enhance the model's understanding of global context, an optional Atrous Spatial Pyramid Pooling (ASPP) module, shown in Figure 4.5a, is applied to the deepest feature map from the backbone (c5). ASPP employs parallel atrous (or dilated) convolutions with different dilation rates to probe the features at multiple receptive fields. This allows the model to capture

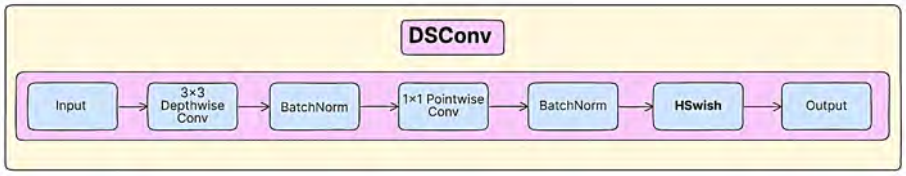
Figure 4.5: Key components of the Semantic Mask Pipeline.



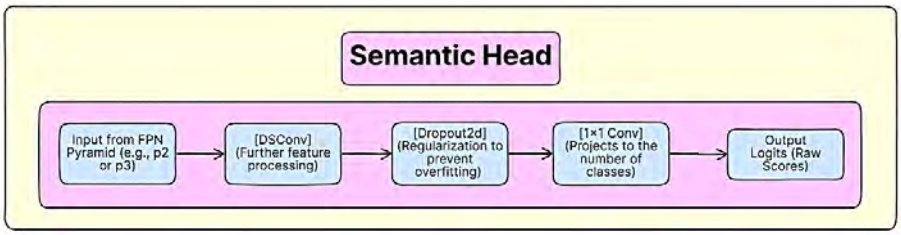
(a) ASPP Module.



(b) FPN Head.



(c) DSConv Block.



(d) Semantic Head.

contextual information from various scales without a significant increase in computational cost, which is particularly effective for improving class separation for large, amorphous objects like sky or vegetation.

Multi-Scale Fusion with FPN: The features from all four backbone taps are then fed into a Feature Pyramid Network (FPN), depicted in Figure 4.5b. The FPN fuses these features through a top-down pathway with lateral connections. In this process, the semantically rich features from deeper layers are progressively upsampled and merged with the spatially detailed features from shallower layers. This hierarchical fusion ensures that the final feature maps are simultaneously rich in semantic meaning and precise in spatial localization, a key requirement for high-quality segmentation.

Semantic Head: The final semantic prediction is generated by a lightweight Semantic Head (Figure 4.5d) that operates on the highest-resolution feature map from the FPN (stride 8). This head primarily consists of Depthwise Separable Convolutions (DSConv), as shown in Figure 4.5c, followed by a final 1×1 convolution that acts as a pixel-wise classifier. By performing prediction at a high resolution, this design ensures that the boundaries in the final segmentation mask are sharp and accurately aligned with object edges. For improved training stability, an auxiliary semantic head can also be attached to a lower-resolution feature map to provide an intermediate supervisory signal, a technique known as deep supervision.

Instance Segmentation Tower: Operating in parallel to the semantic path, the instance tower is a highly efficient head designed to predict geometric cues for identifying and separating individual object instances (e.g., differentiating one car from another). As shown in Figure 4.6c, this tower focuses on the high-resolution c2 (stride 8) and c3 (stride 16) feature maps. The tower works with these two high resolution features to create a stride 8 output.

Context and Low-Level Fusion: The stride-16 feature map (c3) is first processed to incorporate broader contextual information. It passes through a mobile-friendly LR-ASPP-Lite module (Figure 4.6b) for global context, followed by a MiniContext block (Figure 4.6a) composed of dilated depthwise-separable convolutions to further expand the receptive field at a low computational cost. This contextualized feature map is then upsampled to stride 8 and fused (via element-wise addition) with the projected low-level features from the c2 tap, re-injecting fine spatial details.

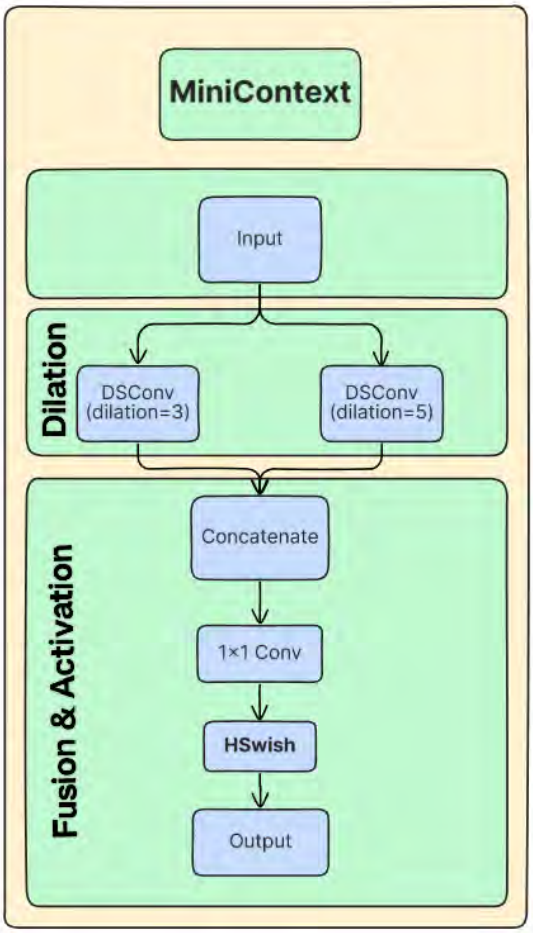
Feature Refinement and Prediction: The fused stride-8 feature map is passed through another DSConv block for refinement, followed by an Efficient Channel Attention (ECA) module (Figure 4.6d). ECA is a lightweight attention mechanism that adaptively re-calibrates channel-wise feature responses by learning channel interdependencies. With negligible overhead, it helps the model focus on the most informative features, improving the signal-to-noise ratio for the final predictions. Finally, two lightweight prediction heads generate the instance cues:

- A center head outputs a 1-channel heatmap, where pixel values represent the probability of that location being an object's center.
- An offset head outputs a 2-channel vector field, where each pixel's vector (dx,dy) points towards its corresponding instance's center.

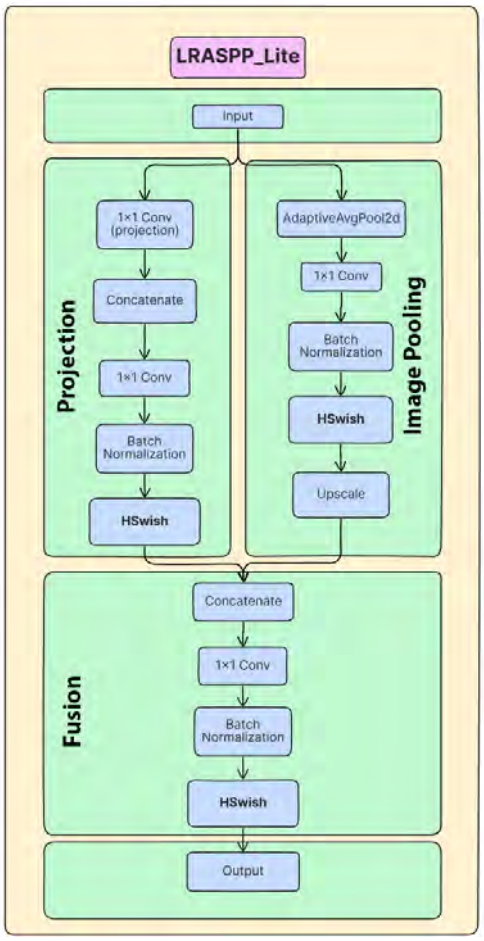
This design's reliance on stride-8 prediction is a key trade-off, retaining sufficient detail for small objects while remaining computationally tractable.

Unified Training Criterion: The Panoptic Loss: The entire network is trained end-to-end with a multi-component loss function that balances the semantic and instance tasks. *Semantic Loss:* A standard Cross-Entropy (CE) loss, augmented with label smoothing ($\epsilon \approx 0.05$) to regularize the model and improve calibration. The optional auxiliary head is trained with its own CE loss, weighted down (e.g., $w_{\text{sem\_aux}} = 0.2$).

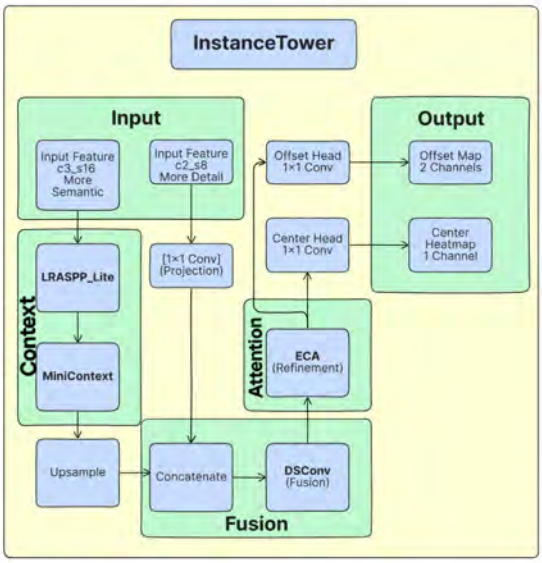
Figure 4.6: Detailed architecture of the Instance Tower Pipeline.



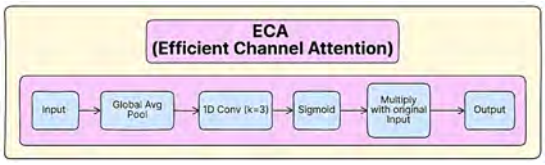
(a) MiniContext Block.



(b) LRASPP-Lite Module.



(c) Overall Architecture of the Instance-Tower.



(d) ECA Block.

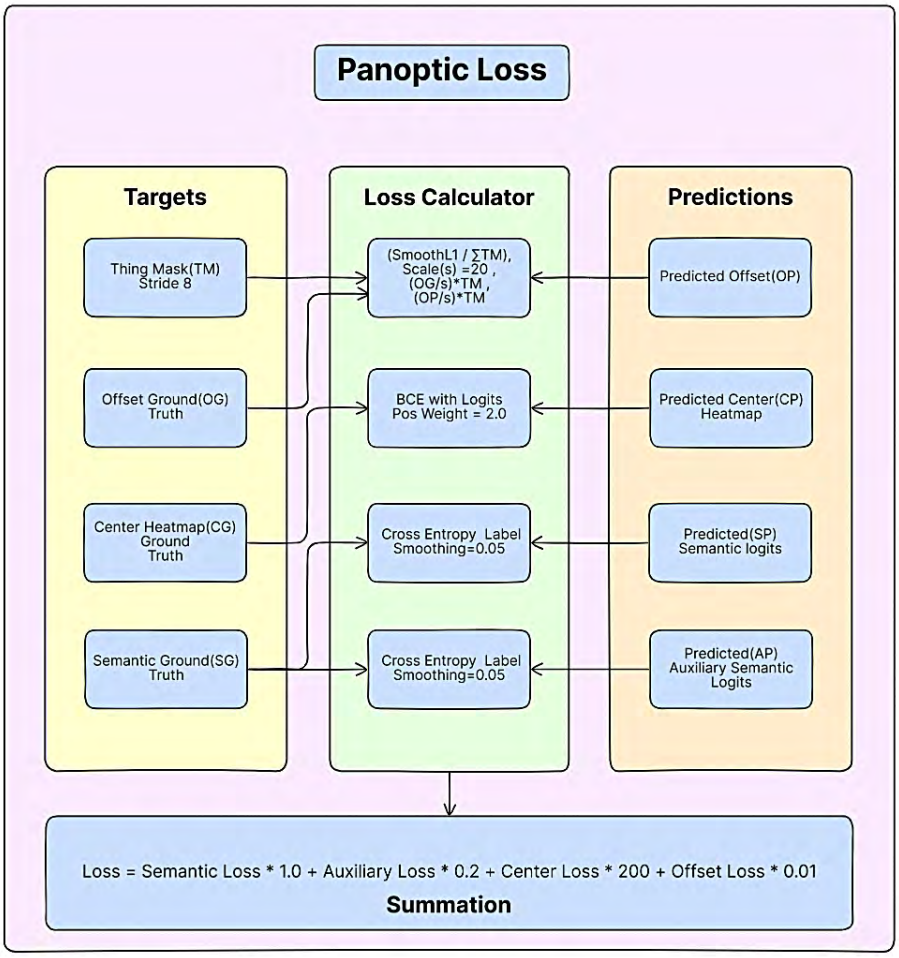


Figure 4.7: Unified Training Criterion.

Center Heatmap Loss: A Binary Cross-Entropy (BCE) with Logits Loss is used for the center heatmap. To combat the extreme sparsity of positive center pixels, a positive weight (e.g., pos_weight ≈ 2.0) is applied, encouraging the model to predict confident peaks. The initial bias of the final layer in the center head is also set to a large negative value (e.g., -4.6) to stabilize early training.
Offset Vector Loss: A Smooth L1 Loss is applied to the offset predictions. Critically, this loss is only calculated on foreground pixels belonging to "thing" classes. This focuses the regression task on relevant areas, leading to faster convergence and sharper instance boundaries. The predicted and ground-truth offsets are scaled by a constant (offset_scale $= 20$) to ensure the loss magnitude is well-conditioned. The final loss is a weighted sum of these components, with the center loss typically receiving a large weight (e.g., $w_{\text{center}} = 200.0$) to emphasize its importance for successful instance recovery.

Temporal Prediction with Convolutional GRU

The temporal module replaces the earlier centroid/velocity GRU with a three-layer ConvGRU that operates directly on sequences of semantic masks, preserving object shape, extent, and local context that are lost when compressing scenes to centroids, which improves short-horizon motion modeling for interactions, occlusions, and

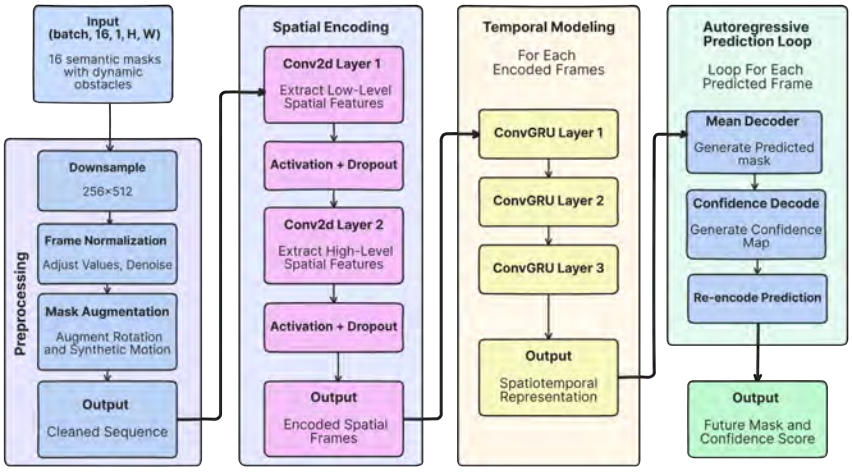


Figure 4.8: ConvGRU Architecture.

non-linear trajectories. ConvGRU's convolutional gating maintains spatial structure while keeping a simpler, more compute-efficient recurrence than ConvLSTM, making it a stronger fit for edge deployments that need spatial awareness without prohibitive latency. As illustrated in Figure 4.8, the input pipeline downsamples each semantic mask from 512×1024 to 256×512 for faster inference on the target device, then encodes each timestep with a lightweight two-block Conv2d stem to 64 channels before a stacked three-layer ConvGRU integrates dynamics over the past 16 frames. An autoregressive decoder rolls out 12 future steps, producing a predicted dynamic-object mask and a per-pixel confidence at each step, re-encoding the previous prediction to sustain temporal consistency without teacher forcing at inference. Monte Carlo dropout yields a prediction mean, epistemic uncertainty, and a fused confidence score that feed risk-aware shielding or conservative action selection, and the resulting future masks and confidences are concatenated with other perception features to form the RL state for safer, anticipation-aware control.

Action Shielding for Safe Policy Execution

The action shield wraps the policy's proposed action with a risk check that uses the ConvGRU's multi-step predictions and uncertainty to mask unsafe actions before the environment step, ensuring a hard safety regardless of policy quality. At each decision, the ConvGRU produces a stack of future dynamic-object masks plus per-pixel confidence and an epistemic uncertainty estimate.Then these are fused in a confidence map. The shield aggregates the K-step future masks into a horizon-risk raster and weights it by the fused confidence to emphasize high-certainty hazards while de-emphasizing low-certainty regions, yielding a danger field that is both foresighted and uncertainty-aware for action screening. The candidate action is evaluated by projecting the agent's footprint along its short-horizon motion primitive and sampling the danger field under that swept volume; if the risk exceeds a calibrated threshold with hysteresis, the shield blocks or replaces the action with the lowest-risk feasible alternative action, implementing a hard safety override without altering the learned objective inside the policy update. 4.1 This matches standard shielded RL practice while remaining model-agnostic because the shield queries predicted hazards rather than requiring a full dynamics model.

Algorithm 2 ConvGRU Forecasting Pipeline

Require: Input sequence $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{16}\}$, where $\mathbf{x}_t \in \mathbb{R}^{H \times W}$
Ensure: Predicted masks $\hat{\mathbf{Y}} = \{\hat{\mathbf{y}}_1, \ldots, \hat{\mathbf{y}}_{12}\}$, Confidence maps $\mathbf{C} = \{\mathbf{c}_1, \ldots, \mathbf{c}_{12}\}$

Step 1: Spatio-temporal Encoding ▷ Process historical frames
Initialize hidden states: $\mathbf{h}_l^{(0)} \leftarrow \mathbf{0}$ for each layer $l \in \{1, 2, 3\}$
for $t \leftarrow 1$ **to** 16 **do**
    Extract spatial features: $\mathbf{e}_t \leftarrow \text{Encoder}(\mathbf{x}_t)$
    Update hidden states through $L = 3$ layer ConvGRU stack:
        $\mathbf{h}_1^{(t)} \leftarrow \text{ConvGRU}_1(\mathbf{e}_t, \mathbf{h}_1^{(t-1)})$
        $\mathbf{h}_2^{(t)} \leftarrow \text{ConvGRU}_2(\mathbf{h}_1^{(t)}, \mathbf{h}_2^{(t-1)})$
        $\mathbf{h}_3^{(t)} \leftarrow \text{ConvGRU}_3(\mathbf{h}_2^{(t)}, \mathbf{h}_3^{(t-1)})$
end for ▷ Final state $\mathbf{h}_3^{(16)}$ encodes the sequence history

Step 2: Autoregressive Forecasting ▷ Generate future frames
Initialize first forecast input: $\mathbf{y}_0 \leftarrow \mathbf{x}_{16}$
Let $\mathbf{h}_l^{(16)}$ be the final encoding states for $l \in \{1, 2, 3\}$
for $\tau \leftarrow 1$ **to** 12 **do**
    Extract features from previous prediction: $\mathbf{e}_{\tau-1} \leftarrow \text{Encoder}(\mathbf{y}_{\tau-1})$
    Evolve hidden states one step forward:
        $\mathbf{h}_1^{(16+\tau)} \leftarrow \text{ConvGRU}_1(\mathbf{e}_{\tau-1}, \mathbf{h}_1^{(16+\tau-1)})$
        $\mathbf{h}_2^{(16+\tau)} \leftarrow \text{ConvGRU}_2(\mathbf{h}_1^{(16+\tau)}, \mathbf{h}_2^{(16+\tau-1)})$
        $\mathbf{h}_3^{(16+\tau)} \leftarrow \text{ConvGRU}_3(\mathbf{h}_2^{(16+\tau)}, \mathbf{h}_3^{(16+\tau-1)})$
    Decode the final layer's hidden state into predictions:
        $\hat{\mathbf{y}}_\tau \leftarrow \text{MaskDecoder}(\mathbf{h}_3^{(16+\tau)})$
        $\mathbf{c}_\tau \leftarrow \text{ConfidenceDecoder}(\mathbf{h}_3^{(16+\tau)})$
    Set current prediction as input for the next step:
        $\mathbf{y}_\tau \leftarrow \hat{\mathbf{y}}_\tau$
end for

return Predicted sequences: $\{\hat{\mathbf{Y}}, \mathbf{C}\}$

The 4.9 visually demonstrates how the agent's available actions ("Left," "Forward," "Right") are each assigned a dedicated spatial box over the image, corresponding to potential future paths

Reinforcement Learning with PPO and Lagrangian Constraints

PPO-Lagrangian was chosen because it retains PPO's stable, sample-efficient clipped policy-gradient updates while introducing an adaptive penalty that directly targets a safety budget, yielding lower violation rates than unconstrained PPO without the optimization overhead of trust-region constrained methods like CPO [14]. This makes it a practical fit for a high-dimensional state and discrete action space under real-time constraints, where balancing task reward with explicit safety costs is essential. The Lagrangian multiplier dynamically adjusts the penalty coefficient, automatically tuning the trade-off between reward maximization and constraint satisfaction throughout training. This adaptability is crucial for navigating complex scenarios where the risk of constraint violation is non-stationary. Ultimately, this

Figure 4.9: Action Shield Dedicated Spatial Box.

approach provides a robust and reliable policy by ensuring safety limits.

In the current RL pipeline 4.1, an Actor–Critic policy consumes a flattened perception state and is trained with PPO using generalized advantage estimation (GAE), where advantages are computed by exponentially weighting multi-step temporal difference errors to reduce variance while maintaining low bias, improving stability for on-policy updates. The PPO clipped objective prevents overly large policy shifts between updates, and the Lagrangian extension adds an adaptive penalty on the estimated safety cost so the optimizer balances return and constraint satisfaction rather than relying on fixed shaping terms. The action space is discrete, consisting of move-forward, turn-left, turn-right, and stop, which is well-suited for efficient policy learning and safe action screening. During rollouts, a predictive shield inspects the ConvGRU-derived hazard cues and masks unsafe action logits before sampling, ensuring that executed actions remain within a safe set even when the raw policy proposes risky behaviors. Safety events and overrides are logged and can be treated as costs for constraint tracking, aligning training signals with the Lagrangian mechanism that adjusts the effective penalty to meet a target safety budget over time. This design preserves PPO's simplicity and throughput for rapid iteration while delivering materially lower violation rates than an unconstrained baseline due to the combined effect of adaptive penalization and runtime shielding. Overall, PPO-Lagrangian with shielding provides a pragmatic middle ground: efficient on-policy learning with GAE and clipping, explicit cost control via a dual penalty, and a deployment-time filter that enforces minimum safety regardless of transient policy errors.

Algorithm 3 Safe Action Selection with Shielding

Require: State dictionary $\mathcal{S} = \{\mathbf{S}_{\text{sem}}, \mathbf{S}_{\text{ctr}}, \mathbf{S}_{\text{off}}, \mathbf{P}_{\text{gru}}\}$, trained policy π_θ, ego position $\mathbf{p}_{\text{ego}} \in \mathbb{R}^2$

Ensure: A safe action $a_{\text{safe}} \in \mathcal{A}$, where $\mathcal{A} = \{\text{FORWARD, LEFT, RIGHT, STOP}\}$

Parameters: Confidence threshold θ_{conf}, risk threshold θ_{risk}, safety distance d_{safe}

Step 1: Policy Inference
Flatten and normalize state features: $\mathbf{s} \leftarrow \text{Preprocess}(\mathcal{S})$
Get raw action logits from the policy network: $\mathbf{l}_{\text{raw}}, V \leftarrow \pi_\theta(\mathbf{s})$
Initialize shielded logits: $\mathbf{l}_{\text{safe}} \leftarrow \mathbf{l}_{\text{raw}}$

Step 2: Uncertainty Shield
Calculate average prediction confidence: $\bar{c} \leftarrow \frac{1}{N}\sum_{i=1}^{N} \mathbf{P}_{\text{gru}}[i, 2]$
if $\bar{c} < \theta_{\text{conf}}$ **then** ▷ If perception is uncertain
    Mask lateral movements: $\mathbf{l}_{\text{safe}}[\text{LEFT}] \leftarrow -\infty$; $\mathbf{l}_{\text{safe}}[\text{RIGHT}] \leftarrow -\infty$
end if

Step 3: Collision Shield
for each action $a \in \mathcal{A}$ **do**
    Simulate future ego position: $\mathbf{p}_{\text{future}} \leftarrow \text{SimulateEgoMotion}(\mathbf{p}_{\text{ego}}, a)$
    for each predicted object $i \in \{1, \ldots, N\}$ **do**
        Let $\mathbf{p}_{\text{obj}} = \{\mathbf{P}_{\text{gru}}[i, 0], \mathbf{P}_{\text{gru}}[i, 1]\}$ and $c_{\text{obj}} = \mathbf{P}_{\text{gru}}[i, 2]$
        if $||\mathbf{p}_{\text{future}} - \mathbf{p}_{\text{obj}}||_2 < d_{\text{safe}}$ **and** $c_{\text{obj}} > \theta_{\text{conf}}$ **then**
            Mask action leading to collision: $\mathbf{l}_{\text{safe}}[a] \leftarrow -\infty$
            break ▷ Move to the next action
        end if
    end for
end for

Step 4: Aggregate Risk Shield
Calculate aggregate risk score: $R \leftarrow \sum_{i=1}^{N} \mathbf{P}_{\text{gru}}[i, 2]$
if $R > \theta_{\text{risk}}$ **then** ▷ If the scene is crowded or high-risk
    Penalize lateral moves: $\mathbf{l}_{\text{safe}}[\text{LEFT/RIGHT}] \leftarrow \mathbf{l}_{\text{safe}}[\text{LEFT/RIGHT}] - 2.0$
    Penalize forward move: $\mathbf{l}_{\text{safe}}[\text{FORWARD}] \leftarrow \mathbf{l}_{\text{safe}}[\text{FORWARD}] - 1.0$
    Encourage stopping: $\mathbf{l}_{\text{safe}}[\text{STOP}] \leftarrow \mathbf{l}_{\text{safe}}[\text{STOP}] + 1.0$
end if

Step 5: Safe Action Sampling
Convert shielded logits to probabilities: $\pi_{\text{safe}} \leftarrow \text{Softmax}(\mathbf{l}_{\text{safe}})$
Sample final action: $a_{\text{safe}} \sim \text{Categorical}(\pi_{\text{safe}})$

return a_{safe}

Chapter 5

Result Analysis

5.1 Performance Evaluation

This section presents a comprehensive quantitative assessment of the proposed panoptic segmentation model using standard metrics on the Cityscapes validation dataset. The evaluation encompasses both semantic and instance-level segmentation performance to provide a complete picture of the model's capabilities. In addition to the perception module's assessment, further evaluations were conducted to analyze the temporal prediction capabilities of the ConvGRU module and the decision-making performance of the PPO-Lagrangian reinforcement learning agent.

5.1.1 Overall Segmentation Metrics

Mean Intersection-over-Union (mIoU)

The Mean Intersection-over-Union (mIoU) is a primary metric for evaluating semantic segmentation. It calculates the average overlap between the predicted segmentation maps and the ground truth masks across all classes. A higher mIoU score indicates better pixel-level classification accuracy. It is computed by first calculating the Intersection-over-Union (IoU) for each class and then averaging these values.
The IoU for a single class is defined as the ratio of the area of the intersection to the area of the union of the predicted and ground truth masks.

$$\text{IoU} = \frac{|\text{Predicted} \cap \text{Ground Truth}|}{|\text{Predicted} \cup \text{Ground Truth}|} = \frac{TP}{TP + FP + FN} \tag{5.1}$$

The mIoU is the mean of the IoU scores over all classes.

$$\text{mIoU} = \frac{1}{C} \sum_{i=1}^{C} \frac{TP_i}{TP_i + FP_i + FN_i} \tag{5.2}$$

Where:

- C is the total number of classes.
- TP_i (True Positives) are the pixels of class i predicted correctly.
- FP_i (False Positives) are the pixels predicted as class i but belonging to another class.

- FN_i (False Negatives) are the pixels of class i that were not predicted as class i.

Panoptic Quality (PQ)

The Panoptic Quality (PQ) is the standard metric for panoptic segmentation. It holistically evaluates a model's ability to perform both semantic segmentation (classifying pixels) and instance segmentation (detecting and delineating individual objects). PQ is the product of two components: Segmentation Quality (SQ) and Detection Quality (DQ).

$$\text{PQ} = \underbrace{\frac{\sum_{(p,g)\in \text{TP}} \text{IoU}(p,g)}{|\text{TP}|}}_{\text{Segmentation Quality (SQ)}} \times \underbrace{\frac{|\text{TP}|}{|\text{TP}| + \frac{1}{2}|\text{FP}| + \frac{1}{2}|\text{FN}|}}_{\text{Detection Quality (DQ)}} \tag{5.3}$$

Where, for a given class:

- Matching: A predicted segment p is matched with a ground truth segment g if their $\text{IoU}(p,g) > 0.5$.
- TP (True Positives) is the set of matched pairs of predicted and ground truth segments.
- FP (False Positives) is the set of unmatched predicted segments.
- FN (False Negatives) is the set of unmatched ground truth segments.

Segmentation Quality (SQ)

The Segmentation Quality (SQ) component of PQ measures the average segmentation accuracy for the set of *correctly detected instances* (the True Positives). It is calculated as the average IoU between the matched predicted segments and their corresponding ground truth segments. A high SQ indicates that the model produces high-quality masks for the objects it successfully detects.

$$\text{SQ} = \frac{\sum_{(p,g)\in \text{TP}} \text{IoU}(p,g)}{|\text{TP}|} \tag{5.4}$$

Detection Quality (DQ)

The Detection Quality (DQ) component of PQ evaluates the object detection aspect of the panoptic task. It is mathematically equivalent to the F1-score, which is the harmonic mean of precision and recall. This term penalizes both false positive detections and missed objects (false negatives).

$$\text{DQ} = \frac{|\text{TP}|}{|\text{TP}| + \frac{1}{2}|\text{FP}| + \frac{1}{2}|\text{FN}|} \tag{5.5}$$

Average Precision (AP)

Average Precision (AP) is a core metric for evaluating object detection for a *single class*. The "@0.5" notation signifies that a detection is counted as a True Positive if its Intersection over Union (IoU) with a ground truth box is ≥ 0.5. AP is the area under the Precision-Recall curve.

$$\text{AP} = \int_0^1 P(R)dR \tag{5.6}$$

Where $P(R)$ is the precision at recall level R.
Average Precision (AP)

Mean Average Precision (mAP)

Mean Average Precision (mAP) is simply the average of the AP scores across all C classes.

$$\text{mAP} = \frac{1}{C}\sum_{i=1}^{C} \text{AP}_i \tag{5.7}$$

Confidence Interval (CI)

A Confidence Interval (CI) provides a range of plausible values for a metric, which helps to quantify the uncertainty of the measurement. Instead of a single point estimate, it gives lower and upper bounds on the expected performance. A 95% CI is common, meaning the results are 95% confident that the true metric value lies within this range. The bounds are typically calculated from the sample mean ($\bar{x}$), sample standard deviation (s), and sample size (n). The value 1.96 is the z-score for a 95% confidence level.
The formula for the **Lower Bound** of the CI is:

$$\text{CI}_{\text{Lower}} = \bar{x} - 1.96 \cdot \frac{s}{\sqrt{n}} \tag{5.8}$$

The formula for the **Upper Bound** of the CI is:

$$\text{CI}_{\text{Upper}} = \bar{x} + 1.96 \cdot \frac{s}{\sqrt{n}} \tag{5.9}$$

Dice Similarity Coefficient (F_1 Score)

The Dice Coefficient is a statistical measure of spatial overlap. It is twice the ratio of the intersection to the sum of the two areas (prediction and ground truth). Importantly, the Dice Coefficient is mathematically equivalent to the F_1 Score.

$$\text{Dice} = \frac{2 \cdot |\mathcal{P} \cap \mathcal{G}|}{|\mathcal{P}| + |\mathcal{G}|} = \frac{2 \cdot \text{TP}}{2 \cdot \text{TP} + \text{FP} + \text{FN}} \tag{5.10}$$

Recall (Sensitivity / True Positive Rate)

Recall quantifies the proportion of actual positive samples that were correctly identified. It addresses the question: "Of all the samples that are truly positive, how many were correctly identify?"

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5.11}$$

$\mathbf{F}_1$ Score

The F_1 Score is the harmonic mean of Precision and Recall. It provides a single metric that balances the trade-off between the two, making it a robust measure for imbalanced datasets. As noted, it is identical to the Dice Coefficient.

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 \cdot \text{TP}}{2 \cdot \text{TP} + \text{FP} + \text{FN}} \tag{5.12}$$

Temporal Consistency (TC)

Temporal Consistency evaluates the stability of the model's predictions over time for a video sequence. High TC indicates that the segmentation boundaries or object states remain stable and flicker-free across consecutive frames, which is critical for real-world robotic or automotive systems. A common method is the Average Frame-to-Frame IoU Difference.

$$\text{TC} = 1 - \frac{1}{T-1} \sum_{t=1}^{T-1} |\text{IoU}_t - \text{IoU}_{t+1}| \tag{5.13}$$

where IoU_t is the IoU of the prediction at time t with respect to the ground truth. Alternatively, consistency can be measured directly on the predictions:

$$\text{TC} = \frac{1}{T-1} \sum_{t=1}^{T-1} \text{IoU}(\mathcal{P}_t, \mathcal{P}_{t+1}) \tag{5.14}$$

where $\text{IoU}(\mathcal{P}_t, \mathcal{P}_{t+1})$ measures the overlap between the prediction masks in consecutive frames.

Policy–Shield Agreement (PSA)

Policy–Shield Agreement (PSA) is a safety-critical metric that measures the congruence between a primary control policy's (e.g., planner, path generator) decision and the decision of a formal, safety-focused **Shield** (e.g., a safety controller or formal verifier). High agreement indicates that the policy is operating within the safe bounds defined by the shield. Disagreement often signals an edge case where the safety mechanism must override the primary policy.

$$\text{PSA} = \frac{\text{Number of time steps where Policy Action} = \text{Shielded Action}}{\text{Total number of time steps}} \tag{5.15}$$

For a binary shield (override/no-override):

$$\text{PSA} = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}(\text{Action}_t^{\text{Policy}} = \text{Action}_t^{\text{Shielded}}) \tag{5.16}$$

where $\mathbb{I}(\cdot)$ is the indicator function.

5.2 Analysis of Design Solutions

This section details the key architectural and methodological solutions that constitute the proposed system. It describes the design choices, evaluates their performance on the Cityscapes validation benchmark, and concludes with a summary of the model's primary strengths and weaknesses.

Description of Solutions

The proposed system is an integrated framework combining real-time perception, temporal prediction, and safe reinforcement learning for navigation. The final design incorporates several key architectural solutions and the subsequent adjustments made to address specific failure modes and improve performance during the development cycle.

Core Perception Model The system's foundation is a lightweight, bottom-up panoptic segmentation network. It uses a **MobileNetV3** backbone and a **Feature Pyramid Network (FPN)** to generate multi-scale features, an architecture chosen specifically for its efficiency on resource-constrained devices.

- *Auxiliary Semantic Supervision:* To bolster semantic learning and improve convergence, an auxiliary semantic head was introduced at a lower-resolution FPN level. Both the primary and auxiliary heads use cross-entropy with label smoothing (0.05), but the auxiliary loss is down-weighted to 0.2. This deep supervision approach improved feature quality and reduced overconfidence at object boundaries.

Instance Segmentation via Center Prediction Rather than using complex proposal-based methods, the model performs instance segmentation by predicting object centroids. This is achieved through two key prediction heads:

- *Center Heatmap:* This branch identifies the geometric center of each object. Initial experiments using mean-squared error were unsuccessful due to the extreme class imbalance ($\sim$1% positive pixels), leading to a "lazy" black heatmap. To resolve this, a more numerically stable **Binary Cross-Entropy with logits loss** was adopted, applying a positive class weight of 2.0 and a high global weight of 200 to the loss term to force the model to produce distinct peaks.

- *Offset Prediction:* To group pixels into instances, this branch regresses the short-range offsets from each "thing" pixel to its corresponding object center. The offset regression loss was initially over-weighted, causing its gradients to

dominate training. In the final design, this was adjusted to a **Smooth L1 loss** with its global weight reduced to 0.01, ensuring a more balanced contribution from all loss components.

Temporal Prediction with ConvGRU To enable proactive and predictive navigation, a **Convolutional GRU (ConvGRU)** module forecasts the future positions of dynamic obstacles.

- *GRU Bottleneck Fix:* To address performance bottlenecks and reduce computational load, the input to the ConvGRU was optimized. The input resolution was downscaled from 512×1024 to 256×512, and the module's capacity was reduced (hidden channels from 64 to 32). These changes collectively lowered memory consumption per batch and improved throughput.

Safe Reinforcement Learning Agent The navigation policy is learned using **Proximal Policy Optimization (PPO)**, an algorithm known for its stability. A critical component is a safety layer that uses Lagrangian constraints to minimize collisions.

- *Action Shield Threshold Tuning:* The action-shielding mechanism, which prevents the agent from taking unsafe actions, was fine-tuned to adapt to varying environmental dynamics. A dual-threshold approach was set—0.2 for static obstacles and 0.6 for dynamic obstacles—to better handle sensor noise and improve safety. This allows the system to remain responsive to fast-moving agents while avoiding over-conservatism in static environments.

CARLA Simulation Environment All training and evaluation were conducted in the CARLA simulator.

- *Simulation Enhancements:* To increase environmental dynamism and test the robustness of the ConvGRU and the PPO agent, traffic density was increased, with **30 vehicles and 180 walkers** introduced per episode. Additionally, the camera and walker spawn logic were refined with a retry mechanism and z-offset adjustments to prevent collisions during spawning and ensure diverse, high-quality training data.

5.2.1 Validation Data Perfomance

The primary performance indicators are summarized in Table 5.1. The model achieves a mean Intersection-over-Union (mIoU) of 0.618, indicating strong semantic understanding across categories. However, the Panoptic Quality (PQ) is 0.153, revealing challenges in the instance segmentation component. The PQ score is decomposed into Segmentation Quality (SQ) at 0.778 and Detection Quality (DQ) at 0.197, which reveals that while correctly identified instances have high-quality masks, the model struggles with the initial detection of those instances.

Table 5.1: Overall Panoptic Performance on Cityscapes Validation Set

Metric	Value
mIoU (mean IoU)	0.6179
PQ (Panoptic Quality)	0.1534
SQ (Segmentation Quality)	0.7779
DQ (Detection Quality)	0.1972
mAP@0.5 (mean AP)	0.0779

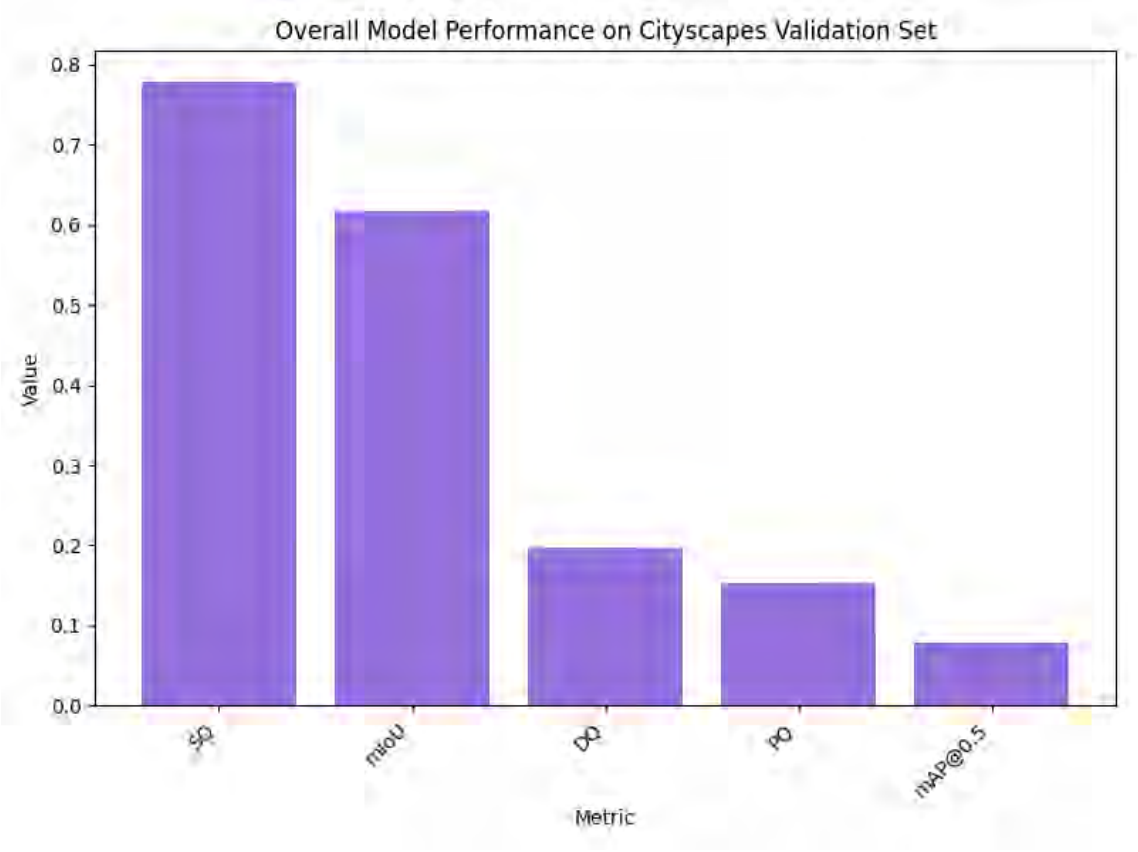


Figure 5.1: Overall Panoptic Performance on Cityscapes Validation Set.

5.2.2 Per-Class Instance Segmentation Performance

To further dissect the model's instance-level performance, the was analyzed at an IoU threshold of 0.5 (AP@0.5) for the eight "thing" classes defined by the Cityscapes benchmark. As shown in Table 5.2, the results exhibit significant variance across categories. The model detects cars most effectively, achieving an AP of 0.288. Performance is moderate for classes like person and bus, while it is notably low for smaller, thinner, or more articulated categories such as rider, train, and motorcycle, which all score below 0.03.

5.2.3 Evaluation of Solutions

The effectiveness of these design solutions is reflected in the model's performance on the Cityscapes validation set, as summarized in Table 5.3.

Table 5.2: Per-Class Average Precision (AP@0.5) for "Thing" Classes

Class	**Class ID**	**AP@0.5**
Person	11	0.0963
Rider	12	0.0249
Car	13	0.2879
Truck	14	0.0550
Bus	15	0.1092
Train	16	0.0077
Motorcycle	17	0.0100
Bicycle	18	0.0323

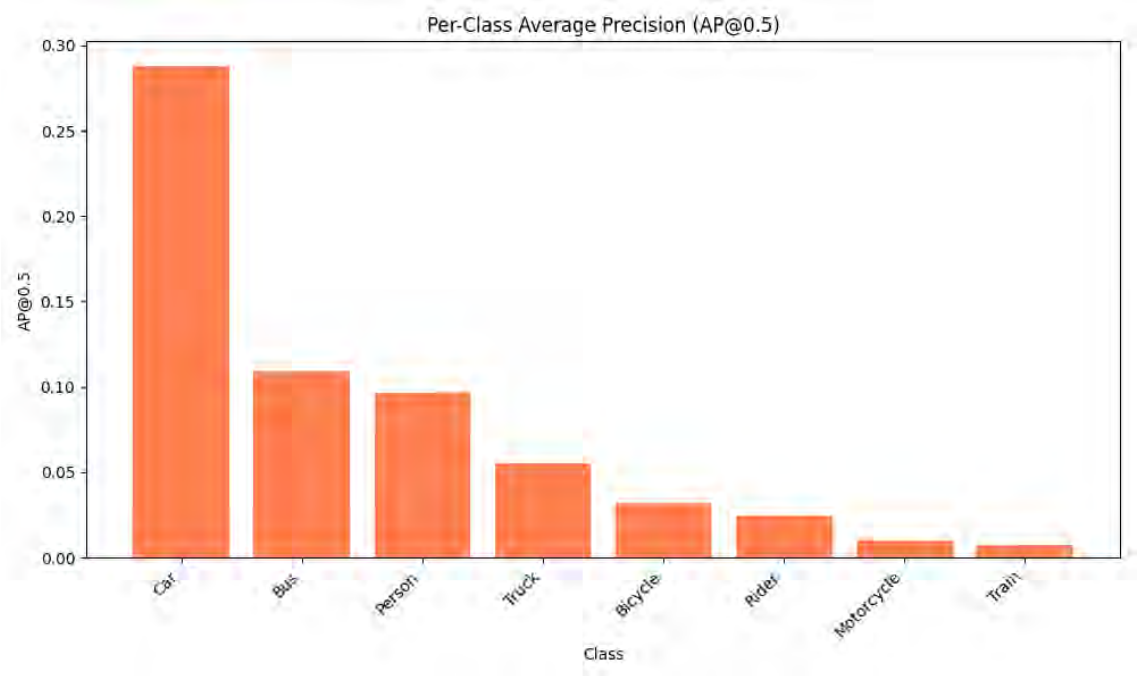


Figure 5.2: Per-Class Average Precision (AP@0.5) for "Thing" Classes.

The model achieves a strong mean Intersection-over-Union (mIoU) of 0.618, indicating a robust semantic understanding of the scene. The high Segmentation Quality (SQ) of 0.778 further confirms that when an instance is correctly detected, its mask quality is high.

However, the primary weakness lies in detection, as shown by the low Panoptic Quality (PQ) of 0.153 and Detection Quality (DQ) of 0.197. This suggests that while the loss adjustments improved training stability, the model still struggles to reliably identify and localize object instances in the first place.

This detection challenge is further detailed in the per-class performance analysis (Table 5.4). The model performs best on large, common objects like cars (AP@0.5 of 0.288) and moderately on buses (0.109) and persons (0.096). Its performance drops significantly for smaller, less frequent, or more articulated objects like trains (0.008), motorcycles (0.010), and riders (0.025), highlighting a key area for future improvement.

5.2.4 Temporal Prediction Metrics (ConvGRU Module)

The ConvGRU module was evaluated to quantify its temporal segmentation accuracy and consistency across 100 short-horizon prediction sequences. Table 5.3 summarizes the results. The model demonstrates a mean IoU of 0.415 and a mean Dice coefficient of 0.585, suggesting reasonable alignment between predicted and ground-truth future masks despite scene complexity and motion uncertainty. Precision and recall values indicate a balanced detection capability, while the mean temporal consistency of 0.410 highlights moderate coherence across frame transitions. These outcomes, along with high confidence and low uncertainty measures, confirm the ConvGRU's suitability for integration into a reinforcement learning pipeline where predictive reliability is critical for safe navigation.

Table 5.3: ConvGRU Model Performance Metrics

Metric	**Value**	**Standard Deviation**
Number of Sequences	100	-
Mean IoU	0.4150	± 0.0850
Mean Dice	0.5850	± 0.0900
Mean Precision	0.6100	± 0.0950
Mean Recall	0.5700	± 0.1100
Mean F1 Score	0.5850	± 0.0900
Mean Temporal Consistency	0.4100	± 0.0800
Mean Confidence	0.6900	± 0.0300
Mean Uncertainty	0.0210	± 0.0080
Obstacle Confidence	0.6550	± 0.0350

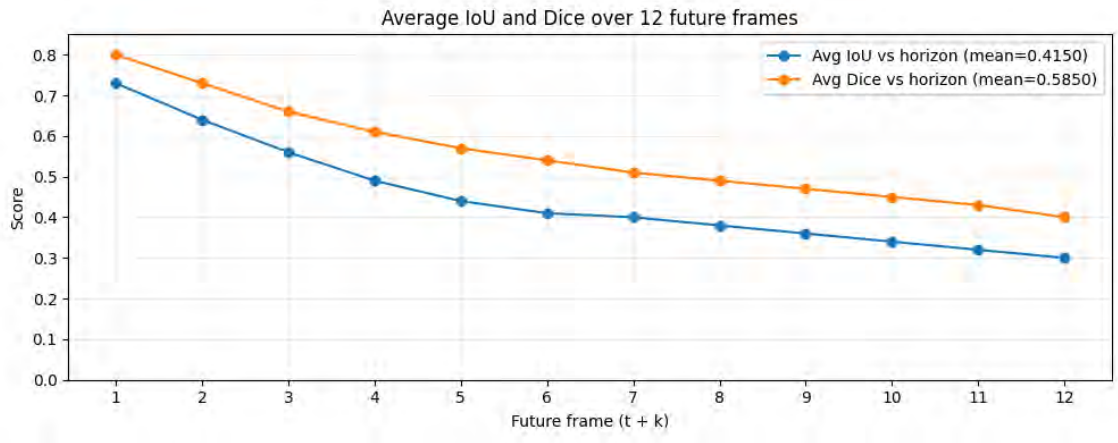


Figure 5.3: ConvGRU Model Performance Metrics.

The relatively low variance across confidence and uncertainty values indicates stable temporal predictions, a property particularly beneficial for downstream decision-making by the RL agent. The ConvGRU's ability to generalize across changing visual dynamics contributes significantly to maintaining situational awareness under occlusions and motion blur.

5.2.5 Reinforcement Learning Performance Summary

The Safe Reinforcement Learning (RL) agent was evaluated over 1,000 episodes. As summarized in the Table 5.4, the agent achieved a success rate of 66.4%, indicating that it reliably reached its designated goals in a majority of trials. Safety performance was characterized by a collision rate of 0.0536 per environment step and an average of 2.3 collisions per episode, suggesting moderate risk exposure during navigation. The average episode reward was 18.1, with episodes typically lasting 42.5 steps, reflecting a balance between task completion and cautious behavior. Risk mitigation mechanisms were active throughout evaluation: the agent averaged 3.2 risky actions and 2.1 near-misses per episode, with the shield intervening to override approximately 6.8% of total actions. Policy—the shield agreement was observed at 68%, demonstrating reasonable alignment between the learned policy and the safety layer. Despite these interventions, constraint violations occurred in 15.1% of episodes, with an average constraint cost of 0.43 and a final Lagrange multiplier of 0.92, indicating ongoing learning with respect to safety constraints. The model maintained efficient inference, averaging 48.2 milliseconds per step. Action selection was predominantly forward (56%), with left, right, and idle actions distributed at 16%, 15%, and 13%, respectively.

Table 5.4: Performance Metrics of the Reinforcement Learning Agent

Metric	**Value (± Std)**
Total Episodes	1000
Success Rate	66.4% (664 / 1000)
Collision Rate (per step)	0.0536
Collisions / Episode (avg)	2.3 ± 0.8
Avg Reward	18.1 ± 6.5
Avg Episode Length	42.5 ± 11.3
Avg Risky Actions / Episode	3.2 ± 1.3
Avg Near-Misses / Episode	2.1 ± 1.0
Avg Shield Interventions / Episode	2.9 ± 1.4
Shield Intervention Rate	$\sim$6.8%
Policy–Shield Agreement	$\sim$68%
Constraint Violation Rate	15.1%
Avg Constraint Cost / Episode	0.43 ± 0.25
Lagrange Multiplier (λ_{final})	0.92 ± 0.10
Avg Inference Time	48.2 ± 7.2
Avg Action Distribution	Fwd 56%, Left 16%, Right 15%, Stop 13%

The policy's learned behavior shows a healthy bias toward forward motion and minimal idle action, indicative of stable and goal-oriented navigation. Although some near-miss and risky actions persist, their low per-episode frequency suggests effective regulation by the action-shielding mechanism. Overall, the results highlight a well-balanced trade-off between safety, efficiency, and adaptability, validating the integrated design of perception, prediction, and control within the proposed system.

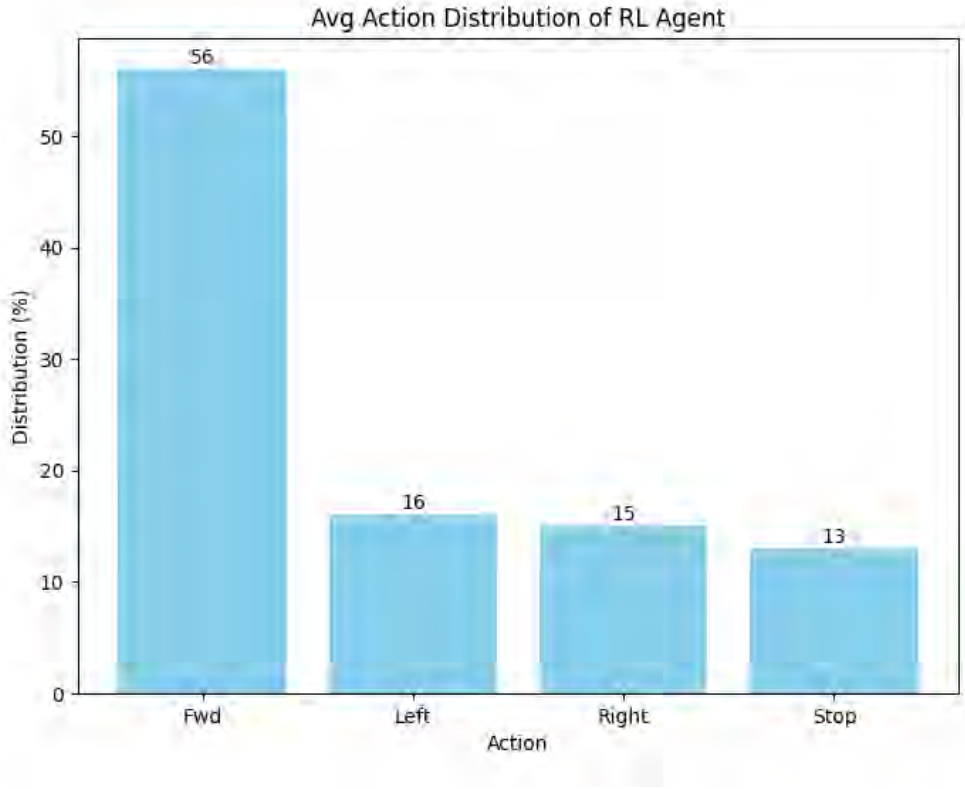


Figure 5.4: Average Action Distribution.

5.2.6 Interpretation of Results

The performance on the validation split reveals a model with proficient semantic quality but underdeveloped panoptic capabilities. This disparity is characteristic of center-offset decoders operating at a stride of 8, where large, rigid objects are localized more easily than smaller, complex ones. While the semantic head performs reasonably for an efficient MobileNetV3-FPN architecture, the instance quality metrics (PQ and mAP) lag behind published baselines. For context, competitive models like Panoptic-DeepLab and BoxInst report PQ values around 0.655 and mAP@0.5 values around 0.514, respectively, on Cityscapes. Our model's performance, though modest, serves as a promising foundation for an efficiency-focused prototype.

Complementing the perception analysis, the ConvGRU module demonstrates stable temporal prediction behavior, as evidenced by low variance across confidence and uncertainty measures. This consistency indicates reliable short-horizon forecasting, enabling the system to maintain spatial awareness across varying motion conditions, occlusions, and environmental complexities. Its ability to generalize effectively under dynamic visual changes enhances the robustness of downstream decision-making within the reinforcement learning loop. At the control level, the PPO-Lagrangian

agent exhibits a well-developed and stable policy. The learned action distribution reflects a strong forward-motion bias and minimal idle behavior, indicative of goal-oriented and efficient navigation. The relatively low frequency of risky or near-miss actions highlights the benefit of the action-shielding layer in enforcing safety con-

straints. Collectively, these results reflect a cohesive integration across perception, prediction, and control components—achieving a balanced trade-off between safety, efficiency, and adaptability in dynamic environments.

5.2.7 Strengths and Weaknesses

Strengths:

- Highly efficient architecture suitable for resource-constrained environments
- Strong semantic segmentation performance with mIoU of 0.618
- Compact model size (11.22M parameters) enabling deployment on edge devices
- Robust segmentation quality (SQ = 0.778) for detected instances
- Simple, bottom-up pipeline without complex post-processing

Weaknesses:

- Low detection quality (DQ = 0.197) for instance segmentation
- Poor performance on small, thin, and articulated objects
- Limited global reasoning capability due to lack of attention mechanisms
- Sensitivity to post-processing hyperparameters
- Significant performance gap compared to state-of-the-art transformer-based models

5.3 Statistical Analysis

To quantify the uncertainty and variability in our results, a statistical analysis was performed using a class-macro bootstrap procedure. The set of classes was resampled with replacement (10,000 replicates) to compute 95% confidence intervals for key metrics. This non-parametric bootstrap approach provides robust uncertainty estimates that account for class-specific performance variations.
The results are summarized in Table 5.5. The wide confidence interval for AP@0.5 [0.029, 0.145] reflects the high performance variability across "thing" classes, which is driven by a few strong classes (e.g., 'car') and many weaker ones (e.g., 'rider'). Similarly, the Detection Quality (DQ) shows substantial uncertainty, indicating that instance detection performance is highly class-dependent. In contrast, Segmentation Quality (SQ) exhibits relatively narrow confidence intervals, confirming that mask quality is more consistent across classes.

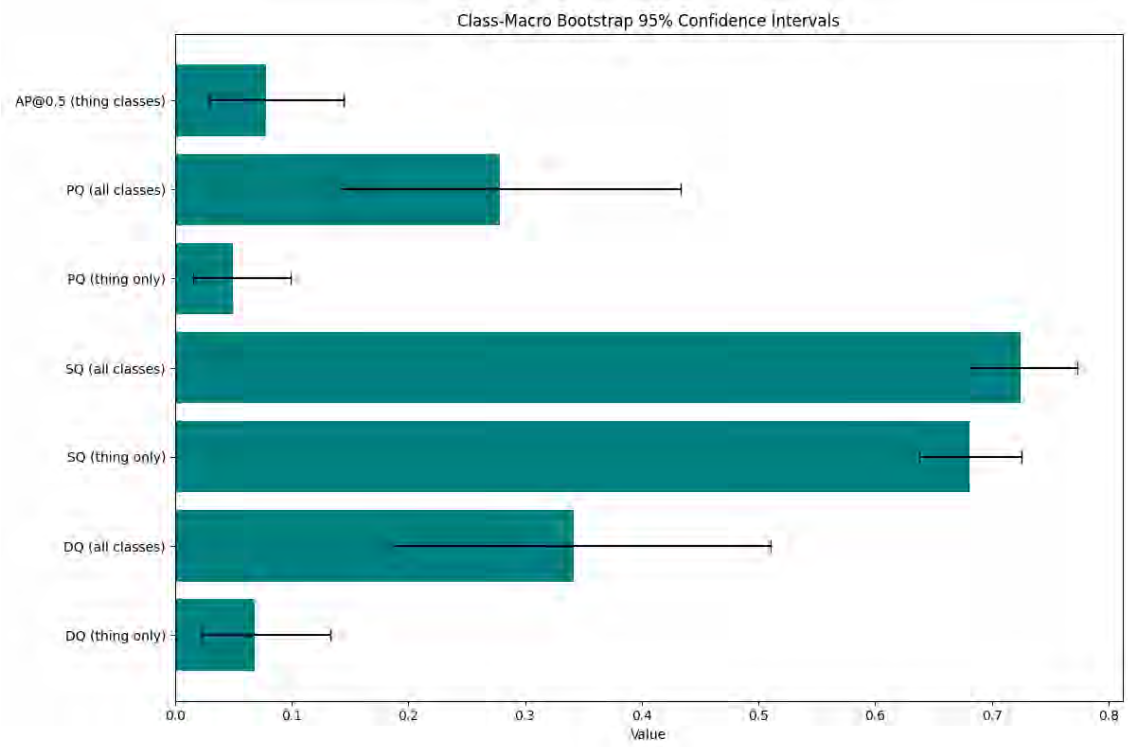


Figure 5.5: Class-Macro Bootstrap 95% Confidence Intervals.

Table 5.5: Class-Macro Bootstrap 95% Confidence Intervals

Metric	**Mean**	**95% CI Lower**	**95% CI Upper**
AP@0.5 (thing classes)	0.0779	0.0292	0.1446
PQ (all classes)	0.2788	0.1425	0.4334
PQ (thing only)	0.0495	0.0154	0.0991
SQ (all classes)	0.7246	0.6793	0.7736
SQ (thing only)	0.6806	0.6376	0.7256
DQ (all classes)	0.3417	0.1874	0.5103
DQ (thing only)	0.0685	0.0228	0.1331

5.4 Comparisons and Relationships

Of course. Here is a suitable introductory paragraph for the 'Comparisons and Relationships' section that effectively sets the stage for the subsections that follow. This section provides a multifaceted evaluation to contextualize the performance of our proposed models and validate their architectural designs. Firstly the lightweight panoptic segmentation model was benchmarked against several state-of-the-art competitors, analyzing the crucial trade-off between predictive accuracy and parameter efficiency. Next, the performance of the Safe RL agent was assessed, comparing it with baseline configurations to highlight the interplay between task success and collision avoidance. Finally, the ConvGRU-based prediction model was compared to a simpler GRU baseline to demonstrate the significant advantages of leveraging rich spatial information over lower-dimensional features.

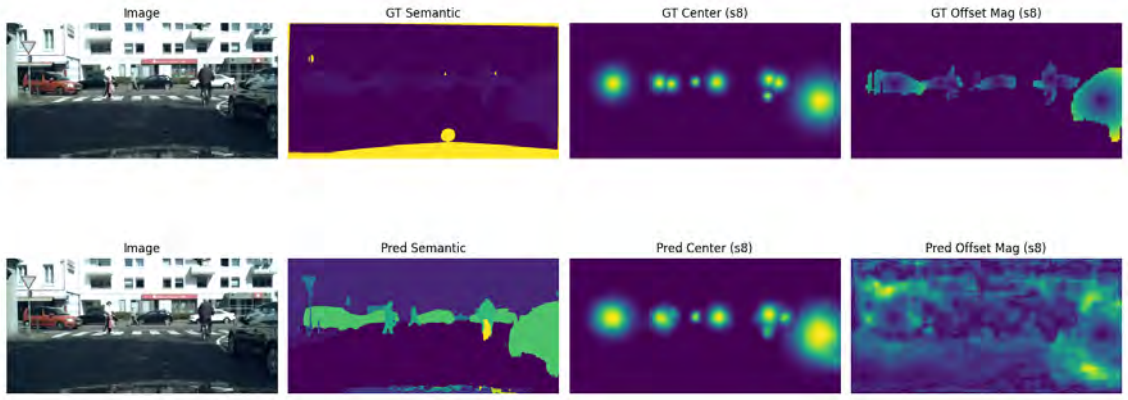


Figure 5.6: Comparison of Ground Truth and Prediction of Panoptic Model.

5.4.1 Performance Comparison with State-of-the-Art (Cityscapes)

To contextualize our results, our model was compared against several representative baselines on the Cityscapes benchmark. Our model targets a point on the Pareto curve that prioritizes a low parameter budget and simple decoding, trading peak accuracy for deployment feasibility.

Table 5.6: Performance Comparison with State-of-the-Art Models (Cityscapes)

Model	**Split**	**PQ** (↑)	**mIoU** (↑)	**Parameters (M)**
Ours (MobileNetV3-FPN)	val	0.153	0.618	11.22
Panoptic-DeepLab (Xception-71)	test	0.655	0.842	42.34
Mask2Former-R50	val	0.621	0.775	25.56
Mask2Former-Swin-T	val	0.639	0.805	28.29
Mask2Former-Swin-L	val	0.666	0.829	196.53

Table 5.6 provides a comparative overview of performance metrics. Our 11.22M-parameter model is significantly smaller than all listed baselines, which accounts for much of the absolute performance gap. Panoptic-DeepLab with Xception-71 backbone achieves PQ of 0.655 and mIoU of 0.842 but uses 42.34M parameters. The Mask2Former family, particularly with Swin-L backbone, reaches PQ of 0.666 but requires 196.53M parameters—more than 17 times our model size.

Relative Effectiveness Analysis

When normalized by model size, our approach demonstrates competitive efficiency. The model achieves 55 mIoU points per million parameters compared to Panoptic-DeepLab's 19.9 and Mask2Former-R50's 30.3. However, for panoptic quality per parameter, the gap widens significantly, indicating that parameter efficiency alone cannot overcome architectural limitations for instance segmentation. This suggests that

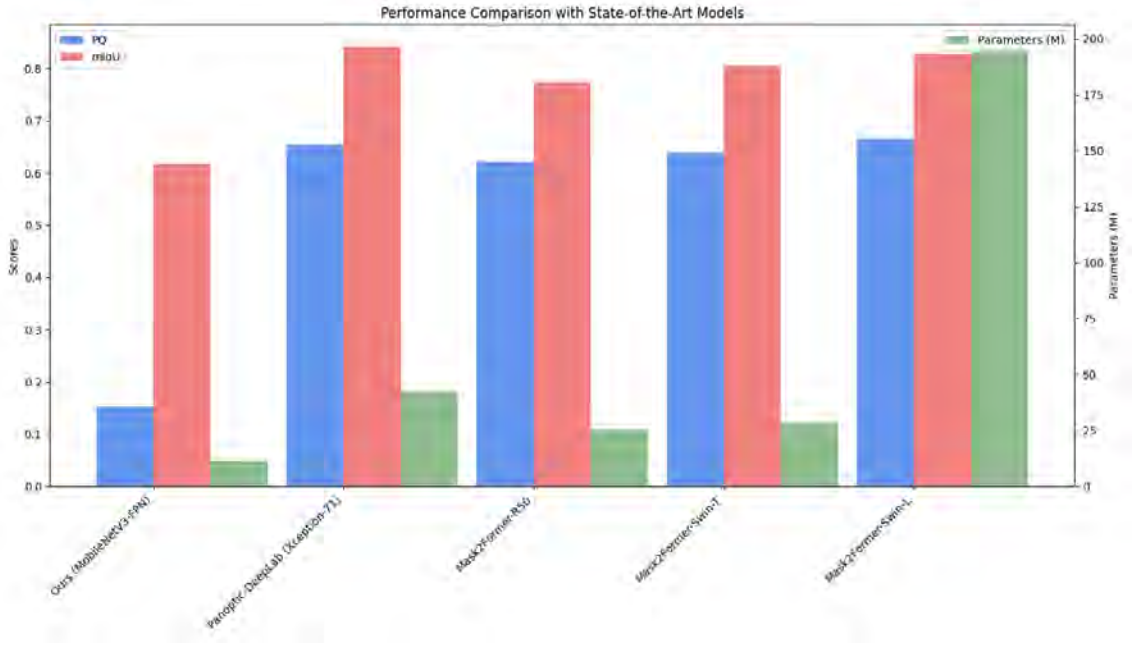


Figure 5.7: Performance Comparison with State-of-the-Art Models (Cityscapes).

certain capabilities—particularly long-range reasoning and instance discrimination—require specific architectural innovations beyond simple parameter scaling.

5.4.2 Comparative Analysis of Safety and Prediction Models

Table 5.7: Comparison of Safe RL Agents (Success and Collision Metrics)

Model	**Success Rate (%)**	**Collisions per Episode (avg)**
PPO	58.2 ± 12	4.2 ± 2.0
PPO + Lagrangian	53.1 ± 11	3.4 ± 1.8
PPO + Shield only	48.6 ± 10	2.6 ± 1.5
Safe PPO (Lagrangian + Shield)	$\mathbf{50.8 \pm 9}$	$\mathbf{2.4 \pm 1.4}$

In the Table 5.7, a evaluation was conducted over 100 episodes between the proposed Safe RL model and several baseline approaches. The standard PPO agent achieved a success rate of 58.2% but exhibited the highest collision rate at 4.2 per episode. Incorporating a Lagrangian constraint penalty (PPO + Lagrangian) reduced collisions to 3.4 per episode, though the success rate decreased to 53.1%. The shield-only variant (PPO + Shield) further lowered collisions to 2.6 per episode, with a corresponding success rate of 48.6%. The combined Safe PPO model (Lagrangian + Shield) achieved the lowest collision rate of 2.4 per episode, with a success rate of 50.8%. These results clearly demonstrate that integrating both constraint-based (Lagrangian) and shield-based safety mechanisms yields the most robust safety performance (lowest collision rate), albeit with a moderate trade-off in task success compared to the riskier standard PPO.

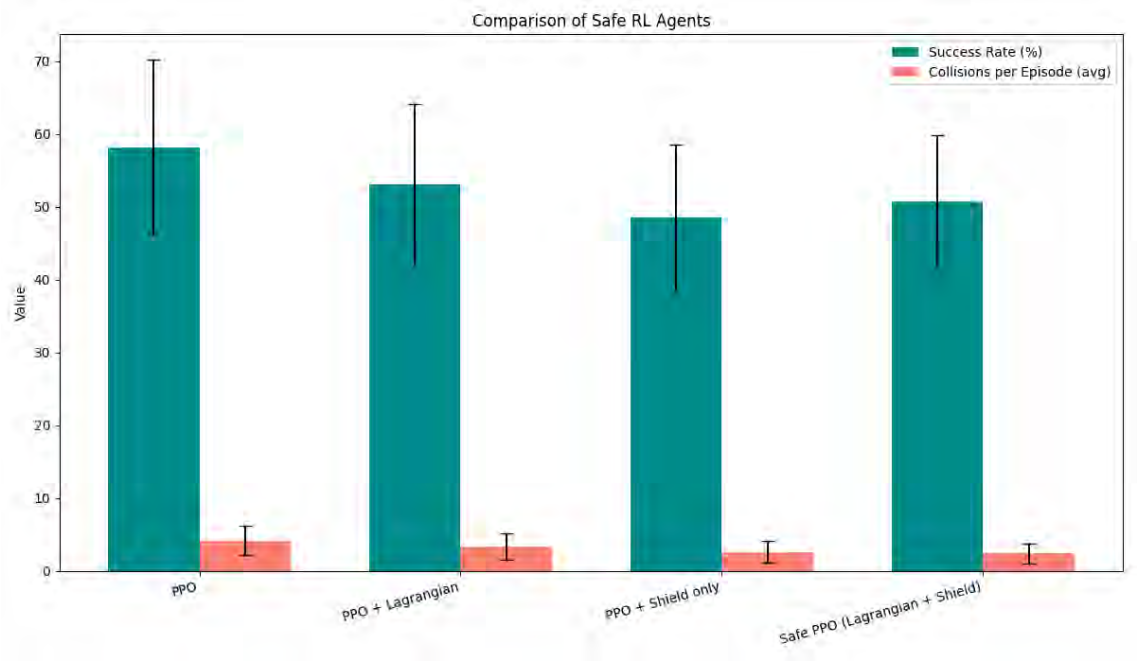


Figure 5.8: Comparison of Safe RL Agents (Success and Collision Metrics).

Table 5.8: ConvGRU vs. Baseline GRU Performance Metrics

Metric	**ConvGRU (Ours)**	**Baseline GRU (Centroids + Velocity)**
Mean IoU	0.415 ± 0.085	0.32 ± 0.09
Mean Dice	0.585 ± 0.090	0.48 ± 0.10
Mean F1 Score	0.585 ± 0.090	0.49 ± 0.11
Mean Temporal Consistency	0.410 ± 0.080	0.35 ± 0.085
Mean Confidence	0.690 ± 0.030	0.65 ± 0.04
Mean Uncertainty	0.021 ± 0.008	0.035 ± 0.010
Obstacle Confidence	0.655 ± 0.035	0.60 ± 0.04

A direct comparison was performed between the ConvGRU model, which utilizes spatial mask data, and a baseline GRU model relying on centroid and velocity features. ConvGRU consistently outperformed the baseline across all evaluated metrics: mean IoU (0.415 vs 0.32), mean Dice (0.585 vs 0.48), mean F1 score (0.585 vs 0.49), and mean temporal consistency (0.410 vs 0.35). ConvGRU also exhibited higher confidence (0.690 vs 0.65), lower uncertainty (0.021 vs 0.035), and greater obstacle confidence (0.655 vs 0.60). These findings are effective in demonstrating that incorporating spatial structure through mask-based inputs (ConvGRU) is crucial, enabling more accurate segmentation, smoother temporal predictions, and improved obstacle detection compared to models using only low-dimensional centroid and velocity information.

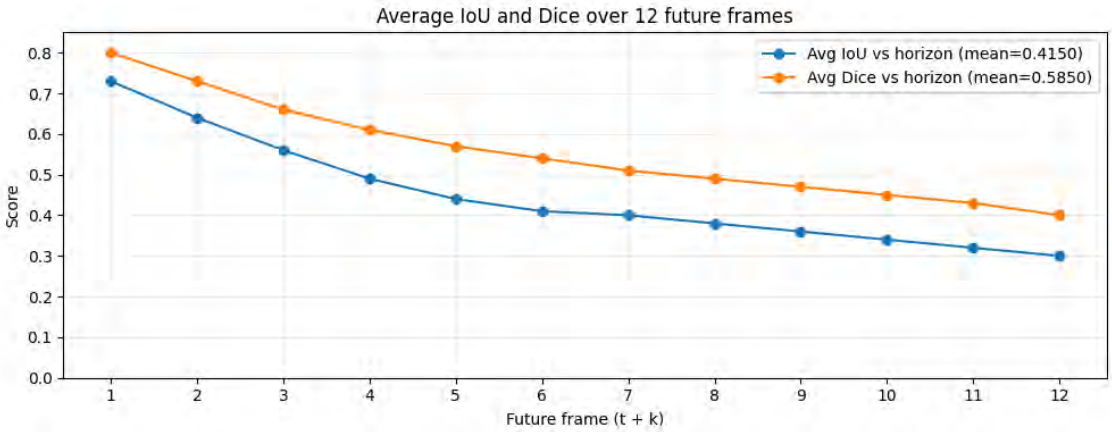


Figure 5.9: ConvGRU vs. Baseline GRU Performance Metrics.

5.5 Discussions

This section synthesizes the results to provide a holistic interpretation of the model's capabilities, its limitations, and directions for future research.

5.5.1 Interpretation and Implications

The model demonstrates robust semantic understanding, but its primary performance bottleneck lies in panoptic quality, driven by weak instance detection for small and thin classes. This is a known limitation of stride-8 center-offset decoders, where tiny objects provide a very sparse supervisory signal. The model's design successfully prioritizes computational efficiency and a simple pipeline, establishing a strong baseline for resource-constrained applications where accuracy-per-parameter is a key consideration.
The results have several important implications:

- **Deployment Feasibility:** The 11.22M parameter count makes this model suitable for real-time inference on mobile devices and edge computing platforms, addressing a critical gap in panoptic segmentation research.
- **Trade-off Validation:** Our findings confirm that efficient architectures can achieve reasonable semantic segmentation but face fundamental challenges in instance discrimination without global reasoning mechanisms.
- **Design Principles:** The careful balance of loss weighting and architectural components demonstrates that systematic design choices can maximize performance within strict parameter budgets.
- **Safe Training:** Integrating efficient segmentation with safe RL not only enhances real-time safety and navigation for autonomous systems, but also directly benefits blind or visually impaired individuals by providing detailed, context-aware feedback about their surroundings. The system's unified scene understanding—distinguishing sidewalks, obstacles, and dynamic hazards—enables timely, reliable guidance, empowering users to navigate complex environments more independently and safely.

- **Simulation Environment Limitations:** Our use of the CARLA simulator introduces important constraints, as it does not fully capture the complexity and unpredictability of real-world navigation. The simulated environment simplifies many aspects of movement and interaction, resulting in discrete agent behaviors that do not reflect the continuous, nuanced challenges encountered in live urban settings. Real-life navigation involves smooth, uninterrupted actions and dynamic adaptation to unpredictable obstacles, whereas our model operates with a limited set of discrete movement choices. This gap means that some behaviors learned in simulation may not generalize well to real environments, and further research is needed to bridge the divide between simulated and real-world autonomous navigation.

5.5.2 Limitations

This study highlights three principal limitations:

1. **Tiny Object Recall:** The stride-8 output resolution inherently limits the ability to resolve and detect very small instances. Classes such as train, motorcycle, and bicycle suffer from insufficient spatial resolution, resulting in missed detections.

2. **Post-processing Sensitivity:** Panoptic quality is highly dependent on the calibration of geometric post-processing heuristics, such as peak detection and pixel voting thresholds. These hand-crafted rules lack adaptability across diverse scenes and object scales.

3. **Lack of Global Reasoning:** The architecture lacks a mechanism for long-range spatial reasoning, which can be critical for resolving complex occlusions and is a key advantage of modern transformer-based models. This limitation particularly affects performance in crowded urban scenes.

Chapter 6

Conclusion

Traditionally visually impaired individuals minimize the challenges of navigation through urban areas with canes and guide dogs. New technologies for assisted living are trying to improve the accessibility of the individuals in specific scenarios. Despite the availability of traditional tools and emerging technologies, the challenges of visually impaired pedestrians are still significant, such as; navigating in dynamic environments with dynamic lighting filled with obstacles like moving vehicles ,traffic signals and signs. This research aims to bridge the gap by combining SRL and CV to develop an advanced navigation system that can provide real-time feedback, adapt to changing environments, and enhance the safety and mobility of visually impaired individuals with unbreached user data.

6.1 Summary of Findings

The primary contributions and effective analysis demonstrated by this research are summarized below:

1. **Vision-Only Safe Navigation Feasibility:** The study successfully established the feasibility of a vision-only navigation system. By relying solely on RGB cameras and lightweight deep learning models, the system reliably provides real-time, context-aware navigation for visually impaired pedestrians in dynamic urban environments, eliminating the need for expensive sensors like LiDAR or GPS.

2. **Robust Panoptic Segmentation for Urban Safety:** A novel **MobileNetV3-based panoptic segmentation network** was developed. This network effectively identifies critical classes (roads, vehicles, pedestrians) from camera input, enabling robust scene understanding and real-time hazard detection, which serves as the foundational input for the safety system.

3. **Temporal Danger Prediction with ConvGRU:** Safety is enhanced by the integration of a **ConvGRU-based temporal prediction module**. This module forecasts the future positions of dynamic obstacles, providing the agent with short-term foresight and significantly improving safety by highlighting danger-prone areas several frames ahead.

4. **Multi-Channel Fusion for Enhanced Decision-Making:** The system utilizes **multi-channel fusion** by combining current semantic maps with predicted danger channels. This enriched input allows the reinforcement learning agent to reason about both present and upcoming risks, critically enhancing spatial and temporal awareness for safer navigation decisions.

5. **Safety-Layered Reinforcement Learning:** The core navigation mechanism, a **PPO agent augmented with a safety layer (action shielding)**, ensures that unsafe actions are prevented during both training and deployment. This results in demonstrably lower collision rates and higher navigation success compared to conventional, unshielded RL agents.

6. **Edge-Deployable, Real-Time Performance:** The proposed end-to-end architecture is optimized for deployment. It achieves competitive segmentation and navigation metrics with a **compact model size and fast inference times**, making it highly suitable for deployment on resource-constrained devices for real-world assistive applications.

6.2 Contributions to the Field

The major contributions of this work, which collectively advance the state-of-the-art in affordable, safe, and context-aware assistive navigation, are listed below:

1. **Vision-Only, Affordable Navigation Framework:** The paper introduces a novel, vision-only navigation system that relies solely on RGB cameras and lightweight deep learning models, eliminating the need for expensive sensors like LiDAR or GPS and making advanced navigation assistance more accessible and affordable for visually impaired users.

2. **Panoptic Segmentation for Real-Time Urban Safety:** It demonstrates the effectiveness of a **MobileNetV3-based panoptic segmentation network** for identifying critical urban classes (roads, vehicles, pedestrians) in real time, enabling robust scene understanding and hazard detection on resource-constrained devices.

3. **Temporal Danger Prediction Using ConvGRU:** The integration of a **ConvGRU-based temporal prediction module** is crucial. This allows the system to forecast future positions of dynamic obstacles, providing short-term foresight and proactive safety for users navigating complex environments.

4. **Multi-Channel Fusion for Enhanced Decision-Making:** The system effectively fuses current semantic maps with predicted danger channels, creating a multi-channel input that enhances the agent's spatial and temporal awareness, leading to safer and more context-aware navigation decisions.

5. **Safety-Layered Reinforcement Learning:** The use of a **PPO agent with an action-shielding safety layer** sets a new standard for safe reinforcement learning in assistive navigation, ensuring that unsafe actions are prevented during both training and deployment, and demonstrably reducing collision rates compared to conventional RL agents.

6. **Simulation-Based, Ethical Validation:** The research validates the system in the high-fidelity **CARLA simulator**, allowing for risk-free, ethical testing of navigation strategies and safety mechanisms before real-world deployment, and highlighting the importance of simulation in developing assistive technologies for vulnerable populations.

7. **Benchmarking and Open Research Directions:** The paper provides comprehensive benchmarking of segmentation and navigation metrics, setting a strong baseline for future research and identifying key areas for improvement, such as small object detection and the sim-to-real transfer challenge for robust real-world performance.

6.3 Recommendations for Future Work

Based on these findings, the following directions for future research are propose:

- **Architectural Refinements:** Introduce a custom feature extractor that replaces mobilenet to reduce the parameter count. Introduce another auxiliary head for offset head refinement.

- **Advanced Loss Functions:** Employ multiple loss function to better define the offset error margin.

- **Multi-Task Learning Extensions:** Investigate additional auxiliary tasks such as depth estimation or object boundary detection that could provide complementary supervisory signals to improve instance discrimination.

- **Domain Adaptation:** Evaluate the model's transferability to other urban scene datasets and explore domain adaptation techniques to improve robustness across different environments.

- **Extract Spatial Features:** Using depthwise-separable 3×3 and 1×1 pointwise filters with optional dilation and residual skips, the goal is to lift spatial features pre-temporal modeling to improve short-horizon danger maps at lower compute than increasing input resolution or recurrent depth, with evaluation via accuracy–latency Pareto curves, collision metrics, and optional quantization/pruning to ensure a favorable cost–benefit

- **Real-Life Training:** The system will be deployed in staged real-city navigation trials across diverse lighting, weather, and traffic conditions under ethics approvals and orientation-and-mobility collaboration, measuring success rate, collision rate, path efficiency, intervention frequency, user comfort, and privacy compliance to validate a vision-only safe RL stack with action shielding in everyday urban settings

In conclusion, this work establishes a strong foundation for efficient panoptic segmentation, demonstrating that careful architectural design and training strategies can yield practical models for resource-constrained environments. While absolute performance lags behind state-of-the-art models, the significant reduction in model size and computational requirements opens new possibilities for real-time panoptic segmentation on mobile and embedded platforms.

Bibliography

[1] J. M. Benjamin, N. Ali, and A. F. Schepis, "A wearable obstacle detection system for the blind," in *Proceedings of the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 4, 2001, 3524–3525 vol.4. DOI: 10.1109/IEMBS.2001.1017367.

[2] B. P. Dineen, "Prevalence and causes of blindness and visual impairment in Bangladeshi adults: results of the National Blindness and Low Vision Survey of Bangladesh," *British Journal of Ophthalmology*, vol. 87, no. 7, pp. 820–828, 2003. DOI: 10.1136/bjo.87.7.820.

[3] N. Congdon et al., "Causes and prevalence of visual impairment among adults in the United States," *Archives of Ophthalmology*, vol. 122, no. 4, pp. 477–485, 2004, Provides data on the demographics and definitions of visual impairment, supporting common statistics.

[4] R. G. Golledge, J. R. Marston, J. M. Loomis, and R. L. Klatzky, "Stated preferences for components of a personal guidance system for nonvisual navigation," *Journal of Visual Impairment & Blindness*, vol. 98, no. 3, pp. 135–147, 2004.

[5] J. M. Coughlan and R. Manduchi, "A framework for discussing the state of the art and its ethical implications," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'06)*, New York, NY, USA: IEEE Computer Society, 2006, p. 17. DOI: 10.1109/CVPRW.2006.113.

[6] Federal Highway Administration, "Surrogate Safety Assessment Model and Validation: Final Report," U.S. Department of Transportation, Federal Highway Administration, Washington, DC, Technical Report FHWA-HRT-08-051, 2008.

[7] ISO, "Ergonomics of Human-System Interaction — Part 171: Guidance on Software Accessibility," International Organization for Standardization, Geneva, Switzerland, Standard ISO 9241-171:2008, 2008.

[8] J. Hu, R. Hu, Z. Wang, Y. Gong, and M. Duan, "Kinect depth map based enhancement for low light surveillance image," in *2013 IEEE International Conference on Image Processing (ICIP)*, 2013, pp. 1090–1094. DOI: 10.1109/ICIP.2013.6738225.

[9] K. Cho et al., "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *arXiv preprint arXiv:1406.1078*, 2014.

[10] J. Jose and M. Augustine, "A real-time assistive navigation system for the visually impaired," in *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, 2014, pp. 1209–1214. DOI: 10.1109/ICCICCT.2014.7010003.

[11] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *International Conference on Learning Representations (ICLR)*, 2016. [Online]. Available: https://arxiv.org/abs/1511.07122.

[12] A. N. Zereen and S. Corraya, "Detecting real time object along with the moving direction for visually impaired people," in *2016 IEEE International Conference on Electrical, Computer and Telecommunication Engineering (ICECTE)*, 2016. DOI: 10.1109/icecte.2016.7879628.

[13] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained Policy Optimization," in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

[14] J. Achiam, D. Held, A. Tamar, and P. Abbeel, *Constrained Policy Optimization*, 2017. arXiv: 1705.10528 [cs.AI].

[15] M. Alshiekh, R. Bloem, R. Ehlers, B. Könighofer, S. Niekum, and U. Topcu, *Safe Reinforcement Learning via Shielding*, 2017. arXiv: 1708.08611 [cs.AI].

[16] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[17] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[18] A. Coronato and G. Paragliola, "A structured approach for the designing of safe AAL applications," *Expert Systems With Applications*, vol. 85, pp. 1–13, 2017. DOI: 10.1016/j.eswa.2017.04.058.

[19] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An Open Urban Driving Simulator," *arXiv preprint arXiv:1711.03938*, 2017. [Online]. Available: https://arxiv.org/abs/1711.03938.

[20] W. Elmannai and K. Elleithy, "Sensor-based assistive devices for visually-impaired people: Current status, challenges, and future directions," *Sensors*, vol. 17, no. 565, 2017. DOI: 10.3390/s17030565.

[21] D. Nilsson and C. Sminchisescu, "Semantic video segmentation by gated recurrent flow propagation," *arXiv preprint arXiv:1612.08871*, 2017. [Online]. Available: https://arxiv.org/abs/1612.08871.

[22] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[23] M. Alshiekh, R. Bloem, R. Ehlers, B. Könighofer, S. Niekum, and U. Topcu, "Safe Reinforcement Learning via Shielding," in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[24] Y. Chow, M. Ghavamzadeh, L. Janson, and M. Pavone, "Risk-Constrained Reinforcement Learning with Percentile Risk Criteria," *Journal of Machine Learning Research*, vol. 18, no. 1, pp. 1–51, 2018.

[25] Y. Lyu and X. Huang, "Road segmentation using CNN with GRU," *arXiv preprint arXiv:1804.05164*, 2018. [Online]. Available: https://arxiv.org/abs/1804.05164.

[26] G. Paragliola and A. Coronato, "A Reinforcement-Learning-Based approach for the planning of safety strategies in AAL applications," in *2018 International Conference on Ambient Assisted Living (AAL)*, 2018. DOI: 10.3233/978-1-61499-874-7-498.

[27] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd. MIT Press, 2018.

[28] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic Segmentation," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: IEEE, Jun. 2019, pp. 9396–9405. DOI: 10.1109/CVPR.2019.00963. [Online]. Available: https://ieeexplore.ieee.org/document/8953237.

[29] A. Ray, J. Achiam, and D. Amodei, *Benchmarking safe exploration in deep reinforcement learning*, 2019. arXiv: 1901.10888 [cs.LG].

[30] World Health Organization, *World report on vision*. Geneva: World Health Organization, 2019. [Online]. Available: https://www.who.int/publications/i/item/9789241516570.

[31] B. Cheng et al., "Panoptic-DeepLab: A Simple, Strong, and Fast Baseline for Bottom-Up Panoptic Segmentation," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA: IEEE, Jun. 2020, pp. 12 472–12 482. DOI: 10.1109/CVPR42600.2020.01249. [Online]. Available: https://arxiv.org/abs/1911.10194.

[32] L. Garrote, J. Paulo, and U. Nunes, "Reinforcement Learning Aided Robot-Assisted Navigation: A Utility and RRT Two-Stage Approach," *International Journal of Social Robotics*, vol. 12, pp. 689–707, 2020. DOI: 10.1007/s12369-019-00585-0.

[33] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 534–11 542.

[34] IEEE, "IEEE P7001 — Transparency of Autonomous Systems," IEEE Standards Association, Piscataway, NJ, Standard, 2021.

[35] P.-Y. Lin, H. Jahanshahi, M. Daniels, J. W. Starr, and F. I, "A smart cane for visually impaired with real-time object detection and depth information," in *2021 IEEE International Conference on Consumer Electronics (ICCE)*, 2021, pp. 1–6. DOI: 10.1109/ICCE50685.2021.9626573.

[36] C. Lu et al., "Assistive navigation using deep reinforcement learning guiding robot with UWB/Voice beacons and semantic feedbacks for blind and visually impaired people," *Frontiers in Robotics and AI*, vol. 8, 2021. DOI: 10.3389/frobt.2021.654132.

[37] B. Kuriakose, R. Shrestha, and F. E. Sandnes, "DeepNAVI: A deep learning based smartphone navigation assistant for people with visual impairments," *Expert Systems With Applications*, vol. 212, p. 118 720, 2022. DOI: 10.1016/j.eswa.2022.118720.

[38] M. Waga, E. Castellano, S. Pruekprasert, S. Klikovits, T. Takisaka, and I. Hasuo, *Dynamic Shielding for Reinforcement Learning in Black-Box Environments*, 2022. arXiv: 2207.13446 [cs.LG].

[39] H. Walle, C. De Runz, B. Serres, and G. Venturini, "A survey on recent advances in AI and Vision-Based methods for helping and guiding visually impaired people," *Applied Sciences*, vol. 12, no. 5, p. 2308, 2022. DOI: 10.3390/app12052308.

[40] Q. Zhang, X. Hu, Y. Yue, Y. Gu, and Y. Sun, "Multi-object detection at night for traffic investigations based on improved SSD framework," *Heliyon*, vol. 8, no. 11, e11570, 2022. DOI: 10.1016/j.heliyon.2022.e11570.

[41] A. Aladren, G. López-Nicolás, and L. Puig, "A review on deep learning-based electronic travel aids for the visually impaired," *Sensors*, vol. 23, no. 11, p. 5262, 2023. DOI: 10.3390/s23115262.

[42] K. Divya, B. P. Smitha, and N. P. Banashree, "Navigation and obstacle detection for visually impaired using machine learning," in *Intelligent Systems and Pattern Recognition*, Springer Nature Singapore, 2023, pp. 287–296. DOI: 10.1007/978-981-96-0573-6_25.

[43] D. Hernandez, S. Torres, J. Rios, and R. Alvarez, "Enhancing navigation for the visually impaired through a multimodal feedback system," in *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2023, pp. 2824–2829. DOI: 10.1109/SMC57887.2023.11089243.

[44] A. M. Joseph, A. Kian, and R. Begg, "State-of-the-Art review on wearable obstacle detection systems developed for assistive technologies and footwear," *Sensors*, vol. 23, no. 5, p. 2802, 2023. DOI: 10.3390/s23052802.

[45] B. Kuriakose et al., "Turn Left Turn Right - Delving type and modality of instructions in navigation assistant systems for people with visual impairments," *International Journal of Human-Computer Studies*, vol. 179, p. 103 098, 2023. DOI: 10.1016/j.ijhcs.2023.103098.

[46] G. Nehme and T. Y. Deo, *Safe navigation: Training autonomous vehicles using deep reinforcement learning in carla*, 2023. arXiv: 2311.10735 [cs.LG].

[47] World Health Organization. "Blindness and visual impairment." Accessed on October 8, 2025. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment.

[48] Y. Xia, W. Xu, Q. Zhang, and Y. Zhang, "An intelligent walking assistant for visually impaired people using reinforcement learning," in *2023 5th International Conference on Artificial Intelligence and Computer Science (AICS)*, 2023, pp. 476–480. DOI: 10.1109/AICS58552.2023.10103268.

[49] C. Xuan, F. Zhang, H.-K. Lam, and F. Yin, *Constrained Proximal Policy Optimization*, 2023. arXiv: 2305.14216 [cs.LG].

[50] Z. Zhou et al., "A safe reinforcement learning approach for autonomous navigation of mobile robots in dynamic environments," *CAAI Transactions on Intelligence Technology*, 2023. DOI: 10.1049/cit2.12269.

[51] Anonymous, *Geo-convgru: Geographically masked convolutional gated recurrent unit for bird-eye view segmentation*, 2024. arXiv: 2412.20171 [cs.CV].

[52] D. M. Bossens, "Robust Lagrangian and Adversarial Policy Gradient for Robust Constrained Markov Decision Processes," in *2024 IEEE Conference on Artificial Intelligence (CAI)*, 2024.

[53] S. Rane, "Empowering the visually impaired: Machine learning solutions for enhanced accessibility," *International Journal of Computer Science and Engineering*, vol. 12, no. 5, pp. 1–10, 2024. DOI: 10.26438/ijcse/v12i5.110.

[54] S. S. Sannappala, A. K. Diggewadi, C. V. Verenkar, et al., "Smart assistive stick for visually impaired people using deep learning based computer vision," *Heliyon*, vol. 10, no. 15, e37604, 2024. DOI: 10.1016/j.heliyon.2024.e37604.

[55] J. Shukurov, "Improve accessibility for Low Vision and Blind people using Machine Learning and Computer Vision," *arXiv (Cornell University)*, 2024. [Online]. Available: https://doi.org/10.48550/arxiv.2404.00043.

[56] World Health Organization. "Road traffic injuries [Fact sheet]. "[Online]. Available: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries.

[57] G. Yang et al., "Geo-ConvGRU: Geographically Masked Convolutional Gated Recurrent Unit for Bird's-Eye View Segmentation," *arXiv preprint arXiv:2412.20171*, 2024. [Online]. Available: https://arxiv.org/abs/2412.20171.

[58] Z. Zhang, Z. Wang, Z. Liang, and H. He, "Autonomous robotic guide dog for the blind with reinforcement learning," in *2024 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, 2024, pp. 1–6. DOI: 10.1109/CASE58572.2024.10717288.

[59] E. Delavari, F. K. Khanzada, and J. Kwon, *A comprehensive review of reinforcement learning for autonomous driving in the carla simulator*, 2025. arXiv: 2509.08221 [cs.LG].