School of Computer Science
FACULTY OF ENGINEERING AND PHYSICAL SCIENCES



UNIVERSITY OF LEEDS

# Final Report

## Robustness Evaluation of SAM2 under Synthetic Adverse-Weather Conditions in Urban Driving Scenes

**Sachin Sindhe**

**Submitted in accordance with the requirements for the degree of BSc Computer Science**

2025/26

COMP3931 Individual Project

The candidate confirms that the following have been submitted.

| **Item** | **Format** | **Recipient(s) and Date** |
|---|---|---|
| Final Report | PDF file | Uploaded to Minerva / Turnitin on 27/04/26 |
| Project code repository | GitHub repository | Shared with supervisor / available on request on 27/04/26 |
| Project artefacts | Organised project folder / supporting files | Full project artefact folder provided separately via OneDrive link by email on 27/04/26 |

The candidate confirms that the work submitted is their own and the appropriate credit has been given where reference has been made to the work of others.

I understand that failure to attribute material which is obtained from another source may be considered as plagiarism.

**Signed:** Sachin Sindhe

**Date:** 27/04/26

© 2025/26 The University of Leeds and Sachin Sindhe

**Summary**

This project evaluates the robustness of Segment Anything Model 2 (SAM2) under synthetic adverse-weather and brightness-related visual corruptions in urban driving scenes. Using a selected Cityscapes subset, five visual corruption conditions were studied: rain, fog, frost, snow, and brightness. Four correspond to adverse-weather conditions, while brightness is treated as a photometric distortion.

SAM2 was applied to clean and corrupted images using a fixed automatic mask-generation setup. Because SAM2 produces class-agnostic masks rather than semantic labels, the primary metric was clean-vs-corrupted foreground consistency Intersection-over-Union (IoU). Supporting metrics included Dice score, best-match mask IoU, mask count difference, and a supplementary ground-truth-based sanity check.

The results show that SAM2 consistency generally decreases as corruption severity increases. Frost caused the strongest degradation, followed by severe snow. Fog and brightness produced more moderate reductions, while rain showed a more scene-dependent and non-monotonic pattern. Overall, the project provides a controlled small-scale evaluation pipeline for studying SAM2 robustness, while acknowledging limitations in dataset size, synthetic corruption generation, and class-agnostic evaluation.

**Acknowledgements**

I would like to thank my supervisor, Dr. Qian Xie, for guidance and feedback throughout the project. I also thank the School of Computer Science for providing the project framework and assessment support.

## Use of Generative AI Tools

I acknowledge the use of generative AI tools during the preparation of this project. ChatGPT (OpenAI, `https://chat.openai.com/`) and Claude (Anthropic, `https://claude.ai/`) were used for support with planning, explanation of concepts, code-review suggestions, report-structure feedback, proofreading, and improving clarity of writing.
Gemini / Nano Banana Pro (Google, `https://gemini.google.com/`) was used as an image-to-image editing tool to generate the final synthetic weather-corrupted images used in the experimental dataset. The generated images were manually reviewed and incorporated into the project pipeline by the author.
All final decisions about the methodology, experiments, analysis, interpretation of results, and report content were made by the author. AI-generated suggestions were checked, edited, and adapted before inclusion.

# Contents


**1 Introduction and Background Research** **1**

1.1 Project Motivation . . . 1
1.2 Aim and Objectives . . . 1
1.3 Research Questions . . . 2
1.4 Background Research . . . 2
1.4.1 Semantic Segmentation for Urban Driving Scenes . . . 2
1.4.2 Robustness under Corruptions and Adverse Weather . . . 2
1.4.3 Foundation Segmentation Models and SAM2 . . . 3
1.4.4 Research Gap and Positioning . . . 4
1.5 Project Scope and Framing . . . 4
1.6 Contributions . . . 5

**2 Methodology** **6**

2.1 Dataset . . . 6
2.1.1 Cityscapes . . . 6
2.1.2 Subset Selection . . . 6
2.2 Weather Corruption Generation . . . 7
2.2.1 Corruption Types . . . 7
2.2.2 Severity Levels . . . 7
2.2.3 Initial Classical Augmentation Pipeline . . . 7
2.2.4 Final High-Quality Image-to-Image Corruption Workflow . . . 8
2.2.5 Severity Control for Image-to-Image Corruptions . . . 9
2.2.6 Prompt Constraints and Scene Preservation . . . 9
2.2.7 File Organisation . . . 10
2.3 Segmentation with SAM2 . . . 11
2.3.1 Model Setup . . . 11
2.3.2 Inference Profile and Fair Comparison . . . 11
2.3.3 Mask Generation Procedure . . . 12
2.3.4 Resumable Inference and Logging . . . 12
2.4 Evaluation Metrics . . . 13
2.4.1 Rationale for Clean-vs-Corrupted Consistency . . . 13
2.4.2 Dense Foreground Map Construction . . . 13
2.4.3 Foreground Consistency IoU . . . 14
2.4.4 Supporting Consistency Metrics . . . 14
2.4.5 Supplementary GT-Based Sanity Check . . . 15
2.4.6 Evaluation Protocol . . . 16
2.4.7 Evaluation Design Choices . . . 16
2.5 Implementation Details . . . 17
2.5.1 Code Structure . . . 17

2.5.2 Validation and Testing . . . 17
2.6 Version Control and Artefact Management . . . 18

**3 Experiments and Results 19**
3.1 Experimental Setup . . . 19
3.1.1 Conditions . . . 19
3.1.2 Compute Resources . . . 19
3.2 Qualitative Results . . . 19
3.2.1 Corruption Grids . . . 19
3.2.2 Segmentation Visualisations . . . 19
3.3 Quantitative Results . . . 21
3.3.1 Foreground Consistency IoU Across Severity . . . 21
3.3.2 Comparison Across Corruption Types . . . 22
3.3.3 Scene Variability . . . 22
3.3.4 Supporting Metrics . . . 23
3.3.5 Supplementary GT-Based Sanity Check . . . 24

**4 Discussion, Limitations, and Conclusion 25**
4.1 Summary of Findings . . . 25
4.2 Discussion . . . 25
4.3 Answers to the Research Questions . . . 26
4.4 Methodological Reflections . . . 26
4.5 Limitations . . . 27
4.6 Methodological Lessons . . . 28
4.7 Future Work . . . 28
4.8 Conclusion . . . 29

**References 30**

**Appendices 32**

**A Self-appraisal 32**
A.1 Critical Self-Evaluation . . . 32
A.2 Legal, Social, Ethical and Professional Issues . . . 33
A.2.1 Legal Issues . . . 33
A.2.2 Social Issues . . . 33
A.2.3 Ethical Issues . . . 34
A.2.4 Professional Issues . . . 34

**B External Materials 35**
B.1 Project Repository and Artefacts . . . 35
B.2 Submitted Artefact Structure . . . 35
B.3 Final Implementation Scripts . . . 37
B.4 Implementation Environment . . . 38
B.5 External Resources Used . . . 38

# List of Figures

2.1 Comparison between early classical augmentation outputs and the final image-to-image corruption workflow. The classical augmentation examples show more artificial overlay-like artefacts, especially for severe snow and fog, while the Nano Banana Pro examples preserve the scene geometry more naturally and produce more plausible adverse-weather conditions. . . . . 8

2.2 Overview of the final experimental pipeline. The workflow starts with a selected Cityscapes subset and synthetic weather corruption generation. SAM2 then performs automatic mask generation under a fixed inference configuration. The resulting masks are used for primary clean-versus-corrupted consistency evaluation, supplemented by a light GT-based sanity check, and finally converted into report assets such as plots, qualitative grids, and segmentation visualisations. The pipeline diagram was created using draw.io [4]. . . . . 11

3.1 Representative corruption grid for `frankfurt_000001_064925_leftImg8bit`. Rows show the clean image and three corruption severity levels, while columns correspond to the five visual corruption types used in this study. . . . . 20

3.2 Representative SAM2 segmentation outputs for `frankfurt_000001_064925_leftImg8bit`, matching the same layout as Figure 3.1. This allows direct visual comparison of how SAM2 output structure changes as corruption severity increases. . . . . 20

3.3 Quantitative robustness results for the five-image evaluation subset. The left plot shows mean foreground consistency IoU across severity levels for each corruption type, while the right plot summarises the Level 5 comparison directly. . . . . 22

B.1 Consistency drop relative to clean output across corruption severity levels. Higher values indicate greater divergence between clean and corrupted SAM2 foreground predictions. . . . . 36

# List of Tables


2.1 Example prompt structure used for rain corruption generation. . . . . 10
2.2 SAM2 automatic mask generation profile used for the final main experiment. . . 12
2.3 Implementation validation checks used in the final project. . . . . 18

3.1 Primary foreground consistency IoU results for the final five-image evaluation. Values show mean ± standard deviation across five selected Cityscapes images. . 21
3.2 Paired change in foreground consistency IoU from Level 1 to Level 5. Positive values indicate lower consistency at Level 5. . . . . 23
3.3 Selected supporting metrics at Level 5. Values are means across the five selected images. . . . . 23
3.4 Supplementary GT-based sanity-check results (mean overlap score across five images). . . . . 24

B.1 Final implementation scripts and their roles. . . . . 37
B.2 Implementation environment used for the final SAM2 experiments. . . . . 38

# Chapter 1

# Introduction and Background Research

## 1.1 Project Motivation

Autonomous driving and intelligent transportation systems depend on reliable scene understanding under a wide range of environmental conditions. A key component of this understanding is image segmentation, which supports downstream tasks such as road interpretation, obstacle reasoning, scene parsing, and spatial awareness. In controlled benchmark conditions, segmentation models can perform well, but real-world deployment is more challenging because visual quality is often degraded by rain, fog, snow, glare, frost, or other forms of adverse weather.

These degradations can reduce contrast, obscure scene structure, introduce texture-like artefacts, and alter the visibility of small objects or fine boundaries. As a result, a segmentation model that appears strong on clean images may behave less reliably when the same scene is corrupted. Evaluating this robustness is important because real systems must continue to operate under distribution shifts rather than only under ideal benchmark conditions.

This project investigates how the Segment Anything Model 2 (SAM2) behaves under synthetic weather corruptions applied to urban driving scenes from Cityscapes. Rather than focusing only on segmentation output in clean conditions, the study asks whether SAM2 remains *consistent* when the same scene is viewed through increasingly severe corruptions.

## 1.2 Aim and Objectives

### Aim

The aim of this project is to evaluate how synthetic weather corruptions affect the robustness of SAM2 segmentation outputs on urban driving imagery.

### Objectives

The project objectives are:

- to generate weather-corrupted versions of selected Cityscapes images across multiple corruption types and severity levels
- to run SAM2 automatic mask generation on both clean and corrupted images under a consistent inference setup
- to quantify robustness primarily through clean-vs-corrupted consistency metrics, including foreground Intersection-over-Union (IoU) rather than relying only on semantic ground-truth comparison
- to analyse which corruption types cause the greatest degradation and whether consistency generally decreases as corruption severity increases

- to support the quantitative findings with qualitative visualisations showing typical failure modes

## 1.3 Research Questions

The project is guided by the following research questions:

- **RQ1:** How does SAM2 segmentation consistency change as corruption severity increases?
- **RQ2:** Which synthetic weather corruption types reduce segmentation consistency the most?
- **RQ3:** What qualitative failure modes appear under adverse corruptions, and do some scene elements appear more affected than others?

## 1.4 Background Research

### 1.4.1 Semantic Segmentation for Urban Driving Scenes

Semantic segmentation is a dense prediction task in which each pixel in an image is assigned to a semantic category. In autonomous-driving perception, this is important because the system must distinguish between road, pavement, vehicles, pedestrians, buildings, vegetation, and other scene elements. Unlike image classification, which produces a single label for the whole image, semantic segmentation requires spatially detailed understanding of the scene.
Early deep-learning approaches such as Fully Convolutional Networks (FCNs) showed that convolutional networks could be adapted for end-to-end pixel-level prediction [12]. Later architectures improved the quality of segmentation by incorporating stronger contextual reasoning and boundary refinement. PSPNet introduced pyramid pooling to capture global scene context [19], while DeepLabv3+ combined atrous spatial pyramid pooling with an encoder-decoder structure to improve object boundaries and multi-scale representation [2]. Instance-level segmentation approaches such as Mask R-CNN also became influential by predicting object masks in addition to bounding boxes [8].
Urban driving datasets provide the basis for evaluating such models in realistic road scenes. Cityscapes is a widely used benchmark for semantic urban scene understanding and provides high-quality pixel-level annotations for street scenes [3]. These annotations make it possible to evaluate conventional semantic segmentation models using metrics such as mean Intersection-over-Union (mIoU). However, this project does not train or benchmark a conventional semantic segmentation network. Instead, it evaluates the robustness of SAM2, a class-agnostic foundation segmentation model, when the visual appearance of Cityscapes images is altered by adverse weather.

### 1.4.2 Robustness under Corruptions and Adverse Weather

Model performance on clean images does not necessarily indicate robustness under real-world visual degradation. Adverse weather can reduce contrast, introduce occlusion, blur image structure, and alter lighting conditions. In autonomous driving, these effects are especially

important because perception systems must remain reliable under rain, fog, snow, frost, glare, and other environmental changes.

The corruption robustness literature provides one way to study this problem systematically. Hendrycks and Dietterich introduced a benchmark for common image corruptions, using multiple corruption types and severity levels to evaluate whether image classifiers remain reliable under distribution shift [9]. Although their work focused on image classification, the idea of evaluating performance across corruption type and severity directly influenced later robustness studies in dense prediction. For semantic segmentation, Kamann and Rother benchmarked robustness with respect to common corruptions using datasets including Cityscapes, PASCAL VOC, and ADE20K [10]. Their work is relevant because it shows that robustness should be evaluated at the level of the full perception task, not only through clean benchmark performance.

Adverse-weather driving datasets also provide an important reference point. ACDC was introduced specifically for adverse-condition driving perception, containing images under conditions such as fog, rain, snow, and nighttime, together with semantic annotations and corresponding normal-condition images [16]. Such datasets demonstrate the importance of evaluating perception systems under realistic environmental degradation. However, real adverse-weather datasets do not always provide controlled severity levels for the same scene. This project therefore uses synthetic weather corruption to create controlled clean-corrupted pairs, while acknowledging that a larger future study could compare the same pipeline on real adverse-weather datasets such as ACDC.

### 1.4.3 Foundation Segmentation Models and SAM2

The Segment Anything Model (SAM) introduced promptable segmentation as a foundation-model approach to image segmentation [11]. Rather than training a model for a fixed set of semantic classes, SAM produces masks in response to prompts such as points, boxes, or automatic mask generation. SAM2 extends this family of models to images and videos, while retaining the core idea of promptable, class-agnostic segmentation [15].

This class-agnostic behaviour is important for the evaluation design of this project. A conventional semantic segmentation model produces class-labelled outputs, allowing direct comparison with semantic ground truth. SAM2 automatic mask generation, by contrast, produces an unordered set of masks without semantic class labels. This means that a direct semantic mIoU comparison does not fully capture the behaviour of the model. For this reason, this project uses clean-vs-corrupted foreground consistency as the primary robustness metric, while retaining a lightweight ground-truth-based overlap check as supplementary evidence.

The most directly related prior work of SAM robustness for autonomous driving in adverse weather is the preprint study by Shan and Zhang [17]. Their work evaluates the original SAM, using normal driving images from BDD100K [18] and synthetic weather corruptions at multiple severity levels. They compare SAM mask outputs with ground-truth masks by selecting the best-overlapping predicted mask and report that segmentation performance decreases under adverse weather, especially under stronger corruptions. Their clean baseline provides useful context for how SAM-style mask generation behaves under normal conditions before corruption is applied.

This project differs from Shan and Zhang in four main ways. First, it evaluates SAM2 rather than the original SAM. Second, it uses Cityscapes rather than BDD100K as the controlled urban driving dataset. Third, it treats clean-vs-corrupted consistency as the primary metric, rather than relying only on a ground-truth best-overlap score. Fourth, it uses a high-quality image-to-image corruption workflow for the final reported corruptions, because early classical augmentation outputs were visually unrealistic in severe cases. Therefore, the project is not simply a repetition of the previous SAM study; it adapts the robustness question to SAM2 and to a consistency-based evaluation protocol that better matches class-agnostic automatic mask generation.

### 1.4.4 Research Gap and Positioning

The reviewed literature suggests that adverse-weather robustness has been studied for conventional semantic segmentation models and, more recently, for SAM. However, there remains a gap in understanding how SAM2 automatic mask generation behaves when the same urban driving scene is exposed to controlled weather corruptions. Since SAM2 does not output semantic labels, its robustness cannot be fully assessed using only standard semantic segmentation metrics. A consistency-based evaluation is therefore useful because it asks whether the model produces stable foreground structure when the image appearance changes. This project addresses that gap through a small-scale controlled evaluation. It does not aim to provide a benchmark-scale robustness study, but instead develops and analyses a reproducible pipeline for testing SAM2 under synthetic adverse-weather corruptions. The study contributes evidence on which weather types most disrupt SAM2 output stability and clarifies the methodological challenges involved in evaluating class-agnostic foundation segmentation models in autonomous-driving scenes.

## 1.5 Project Scope and Framing

The project focuses on a controlled, small-scale evaluation rather than a full benchmark-scale study. Five selected Cityscapes images were used consistently across four adverse-weather corruption types (rain, fog, snow, and frost) and one photometric brightness distortion and three severity levels. This design was chosen to keep the experiments computationally manageable while still allowing systematic comparison across conditions. The study therefore prioritises methodological clarity and reproducibility over dataset scale.
The final evaluation is framed primarily as a *consistency* study:

- clean image → SAM2 masks
- corrupted version of the same image → SAM2 masks
- compare the two outputs using overlap-based consistency metrics

This framing is important because the main reported result is not semantic accuracy against ground truth, but segmentation robustness under visual corruption. The project still retains a GT-based route as supplementary material, but the primary dissertation narrative is based on clean-vs-corrupted consistency.

## 1.6 Contributions

The main contributions of this project are:

- A controlled evaluation pipeline for applying SAM2 automatic mask generation to clean and weather-corrupted Cityscapes images.
- A documented high-quality corruption-generation workflow using image-to-image generation for rain, fog, frost, snow, and brightness at three severity levels.
- A clean-vs-corrupted foreground consistency evaluation protocol designed for class-agnostic SAM2 masks, supported by Dice score, best-match mask IoU, mask count difference, and a supplementary GT-based sanity check.
- A set of qualitative and quantitative analyses showing how SAM2 output stability changes under different corruption types and severity levels.
- A critical discussion of the limitations of the study, including small dataset scale, synthetic corruption generation, runtime constraints, and the distinction between foreground consistency and semantic segmentation accuracy.

Overall, the project contributes a practical framework for studying robustness of a class-agnostic foundation segmentation model under synthetic adverse weather, with an emphasis on reproducibility, interpretability, and controlled comparison.

# Chapter 2

# Methodology

## 2.1 Dataset

### 2.1.1 Cityscapes

This project uses images from the Cityscapes dataset, a widely used benchmark for urban-scene understanding in autonomous driving research. Cityscapes was preferred for this controlled study because its high-resolution images and established urban-scene annotations make it suitable for detailed mask-based analysis. BDD100K and ACDC are also relevant driving datasets, but Cityscapes provided a clearer controlled baseline for the clean-vs-corrupted comparison used in this project. It also provides high-resolution street-view images collected in real traffic environments and is commonly used for segmentation tasks because of its realistic urban content, including roads, vehicles, pedestrians, poles, signs, and buildings. In the original dataset, images are provided at a resolution of $1024 \times 2048$ pixels. Although Cityscapes also includes fine-grained ground-truth annotations, the final evaluation protocol in this project does not use ground truth as the primary comparison target. Instead, clean-image SAM2 outputs are used as the reference for robustness analysis [3].

### 2.1.2 Subset Selection

Due to the computational cost of automatic mask generation with SAM2 under multiple corruption conditions, the final controlled quantitative evaluation was conducted on five selected Cityscapes images. These images were used consistently across all corruption types and severity levels:

- `frankfurt_000000_011074_leftImg8bit.png`
- `frankfurt_000001_032018_leftImg8bit.png`
- `frankfurt_000001_064925_leftImg8bit.png`
- `frankfurt_000001_065850_leftImg8bit.png`
- `frankfurt_000001_080091_leftImg8bit.png`

This yielded 75 clean-corrupted evaluation pairs across five corruption types and three severity levels. Three of these scenes were also selected as representative qualitative examples for visual comparison figures. Although the subset is still small, it provides a controlled and computationally feasible basis for analysing SAM2 robustness under adverse weather.

## 2.2 Weather Corruption Generation

### 2.2.1 Corruption Types

Five visual corruption conditions were used in the final experiment: rain, fog, frost, snow, and brightness. Four of these correspond to adverse-weather conditions, while brightness is included as a photometric distortion that represents exposure or illumination change. This distinction is important because brightness does not introduce a weather mechanism in the same way as rain, fog, frost, or snow, but it is still relevant for robustness because overexposure and lighting variation can alter contrast, texture, and object boundaries.
The purpose of this stage was not only to make the images visually different, but to test whether SAM2 produces stable segmentation outputs when the appearance of the same underlying scene changes. For this reason, preserving the original scene structure was important: object positions, camera viewpoint, road layout, pedestrian locations, and vehicle placement needed to remain as consistent as possible across clean and corrupted versions.

### 2.2.2 Severity Levels

The final experiment used three severity levels: Level 1, Level 3, and Level 5. These were used to represent light, medium, and strong corruption. The use of severity levels follows the general corruption-robustness idea used in prior work, where model behaviour is examined as corruption strength increases [9, 10, 17]. This project used Levels 1, 3, and 5 rather than all five levels in order to represent light, medium, and strong degradation while keeping SAM2 inference computationally feasible. An earlier version of the project considered five severity levels, but this was reduced to three representative levels following supervisor feedback. This made the experiment more tractable while still allowing the analysis to show whether robustness changed as corruption strength increased.

### 2.2.3 Initial Classical Augmentation Pipeline

An initial corruption-generation pipeline was implemented using conventional image augmentation operations. The script `apply_weather.py` generated rain, snow, fog, brightness, and frost corruptions across five levels, using Albumentations for several effects and a custom overlay for frost. This provided a useful early prototype because it was fast, reproducible, and easy to apply to multiple images. The script also saved outputs into structured condition and severity folders, which helped establish the later experimental directory design. [1]
However, the visual quality of the generated corruptions was not always suitable for the final robustness study. In particular, severe snow and fog often appeared artificial, with overlay-like artefacts that did not resemble realistic adverse weather. This was problematic because the aim of the project was to assess SAM2 behaviour under plausible visual degradation, not only under synthetic pixel-level effects. If the corruption looks unrealistic, then the measured degradation may partly reflect artefacts of the augmentation method rather than meaningful robustness behaviour.



(a) Classical augmentation: snow L5



(b) Nano Banana Pro: snow L5



(c) Classical augmentation: fog L5



(d) Nano Banana Pro: fog L5

Figure 2.1: Comparison between early classical augmentation outputs and the final image-to-image corruption workflow. The classical augmentation examples show more artificial overlay-like artefacts, especially for severe snow and fog, while the Nano Banana Pro examples preserve the scene geometry more naturally and produce more plausible adverse-weather conditions.

### 2.2.4 Final High-Quality Image-to-Image Corruption Workflow

For the final reported experiments, the corrupted images were generated using Nano Banana Pro through the Gemini web/app interface. Nano Banana Pro refers to Google's high-quality Gemini image-generation and image-editing model family. Google's documentation identifies Nano Banana Pro with Gemini 3 Pro Image Preview, which is designed for higher-quality image generation and editing [6, 5]. In this project, it was used as an image-to-image editing tool: the clean Cityscapes image was supplied as the source image, and the prompt instructed the model to preserve the original scene geometry while applying a specified weather condition and severity.

This approach was selected after an initial classical augmentation pipeline produced visually unrealistic severe weather examples. The early `apply_weather.py` pipeline was useful for prototyping because it was fast, deterministic, and easy to organise into condition and severity folders. However, severe snow and fog often appeared as artificial overlays rather than plausible weather transformations. Since the aim of the project was to evaluate SAM2 under visually plausible adverse weather, the final experiment used the higher-quality image-to-image workflow instead.

The final corruption workflow should therefore be understood as a controlled image-editing stage rather than a fully deterministic augmentation process. Control was maintained by using the same clean source images, applying consistent prompt constraints, preserving filenames, manually checking the generated outputs, and organising all images into the same clean/corruption/severity structure required by the evaluation scripts.

Figure 2.1 illustrates the motivation for this design choice. The difference was most visible for

high-severity snow and fog, where the classical method produced strong artificial overlays while the final image-to-image workflow generated more coherent weather transformations. For this reason, the final quantitative and qualitative results in this report are based on the Nano Banana Pro corruption set rather than the earlier augmentation-generated images. Although this image-to-image workflow improved visual realism, it also introduced practical constraints. Generation was slower than classical augmentation and sometimes produced outputs at a different spatial resolution from the original Cityscapes images. These issues were handled during preprocessing and evaluation, while their implications are discussed further in Chapter 4.

### 2.2.5 Severity Control for Image-to-Image Corruptions

The final image-to-image corruption set used three severity levels: Level 1, Level 3, and Level 5. These were defined qualitatively as light, moderate, and severe corruption. Since the final images were generated through the Gemini web/app interface rather than a deterministic augmentation library, severity was controlled through prompt wording rather than fixed numeric parameters.

For each corruption type, the prompts specified how severity should affect visible properties such as density, opacity, visibility distance, contrast reduction, streak intensity, haze, surface wetness, or brightness change. For example, the rain prompts used sparse thin rain streaks and mild wetness for Level 1, increased streak density and road sheen for Level 3, and dense rain streaks with stronger haze and reflections for Level 5. Similar qualitative severity descriptions were used for the other corruption types.

The use of Levels 1, 3, and 5 follows the general idea of corruption robustness studies that evaluate models at increasing severity levels [9, 10, 17]. However, this project uses three representative points rather than all five severity levels, so the results should not be interpreted as a direct replication of five-level benchmark protocols. The three selected levels were intended to represent light, medium, and strong degradation while keeping the SAM2 inference workload feasible.

### 2.2.6 Prompt Constraints and Scene Preservation

To reduce the risk of generating a different scene, the image-to-image prompts followed a consistent set of preservation constraints. The exact wording varied by weather type and severity level, but the same core principles were used throughout the final corruption set.

An example rain-generation prompt is summarised in Table 2.1. The prompt first specified global preservation constraints, then separately defined Level 1, Level 3, and Level 5 rain effects. This helped make the severity progression more consistent across images.

Table 2.1: Example prompt structure used for rain corruption generation.

| **Prompt part** | **Summary of instruction** |
|---|---|
| Global constraints | Preserve camera view, scene geometry, object positions, vehicles, people, image size, crop, and daytime setting. Apply only weather effects and lighting changes consistent with the target weather. |
| Rain L1 | Add sparse thin rain streaks, mild road wetness, subtle reflections, and slightly reduced contrast while keeping the scene clear and daytime. |
| Rain L3 | Increase rain streak density and thickness, add visible wet sheen on road and cars, mild ground spray or haze, and moderate contrast reduction. |
| Rain L5 | Add dense rain streaks, stronger atmospheric haze and spray, reduced but readable visibility, and stronger wet reflections while avoiding an unrealistic night-time effect. |

After generation, the final corrupted images were visually checked against their clean inputs. The checks focused on whether the main camera viewpoint, road layout, vehicles, pedestrians, buildings, and overall framing remained consistent. The generated images and outputs were also shared with the supervisor for review during the project. However, this was not a formal multi-rater annotation process, so the manual check should be viewed as a practical quality-control step rather than an objective validation metric.
No pixel-level similarity metric such as PSNR or SSIM was used to select the final corrupted images. These metrics can be useful for measuring low-level similarity, but they are not ideal for judging weather realism because realistic fog, snow, rain, frost, and brightness changes are expected to alter pixel intensities and local texture substantially. In this project, the more important requirement was structural consistency: the same scene, camera view, and major objects should remain present while the weather appearance changes. A future version could combine manual inspection with structural similarity or feature-based alignment checks.

### 2.2.7 File Organisation

The final images were organised using consistent filenames and folder paths. Each corrupted image kept the same base filename as its clean Cityscapes source image, allowing clean images, corrupted images, SAM2 masks, and ground-truth labels to be matched deterministically. The final input structure followed the pattern `data/clean/` and `data/<condition>/level_<severity>/`. The same structure was used for rain, fog, frost, snow, and brightness.

## 2.3 Segmentation with SAM2

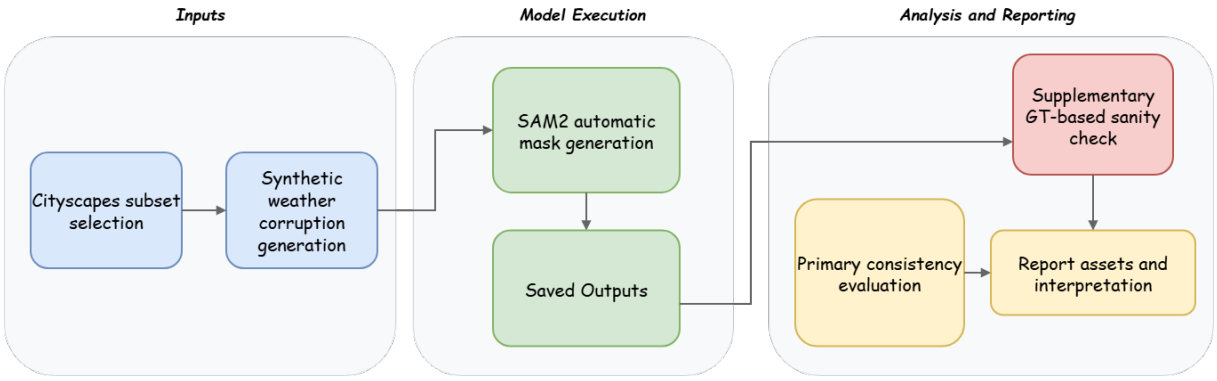



Figure 2.2: Overview of the final experimental pipeline. The workflow starts with a selected Cityscapes subset and synthetic weather corruption generation. SAM2 then performs automatic mask generation under a fixed inference configuration. The resulting masks are used for primary clean-versus-corrupted consistency evaluation, supplemented by a light GT-based sanity check, and finally converted into report assets such as plots, qualitative grids, and segmentation visualisations. The pipeline diagram was created using draw.io [4].

### 2.3.1 Model Setup

Segmentation masks were generated using the Segment Anything Model 2 (SAM2) automatic mask generator [15]. The final experiments used the large SAM2 checkpoint, `sam2.1_hiera_large.pt`, together with the corresponding `sam2.1_hiera_l.yaml` configuration file from the official SAM2 implementation [15, 13]. SAM2 was applied in automatic mask generation mode, meaning that the model produced a set of candidate masks for each image without requiring manual prompts such as points or bounding boxes.
This mode was appropriate for the project because the aim was to evaluate the stability of SAM2 output under adverse weather, rather than to optimise segmentation for a specific manually selected object. Each clean or corrupted image was therefore treated in the same way: it was passed to the SAM2 automatic mask generator and the resulting set of masks was saved for later analysis.

### 2.3.2 Inference Profile and Fair Comparison

A key design decision was to use the same SAM2 mask-generation settings for both clean and corrupted images. This was important because the project measures the effect of weather corruption on SAM2 output. If different mask-generation parameters were used for clean and corrupted images, any observed difference could be caused by the parameter change rather than by the corruption itself.
The final inference script therefore used a shared profile for clean and corrupted inputs. The main reported experiments used the `balanced` profile, which was selected as a compromise between output quality and runtime. Earlier pilot runs showed that heavier settings, such as denser point sampling and crop-based mask generation, could improve visual coverage but were too slow for the full multi-condition experiment. The balanced profile allowed the final five-image, five-corruption, three-severity experiment to remain computationally manageable

Table 2.2: SAM2 automatic mask generation profile used for the final main experiment.

| Parameter | Value |
|---|---|
| Profile name | `balanced` |
| `points_per_side` | 32 |
| `pred_iou_thresh` | 0.86 |
| `stability_score_thresh` | 0.92 |
| `crop_n_layers` | 0 |
| `crop_n_points_downscale_factor` | 2 |
| `min_mask_region_area` | 100.0 |
| `box_nms_thresh` | 0.70 |

while still producing useful qualitative and quantitative outputs. Although the `high_quality` profile produced visually strong masks in early tests, it was not practical for the full final experiment. In pilot runs, high-quality inference commonly required around 1500–2000 seconds per image, with some severe corrupted images taking approximately 2500–3000 seconds. This made it unsuitable for processing multiple images across five corruption types and three severity levels within the available project time. By contrast, the `balanced` profile reduced the runtime substantially while preserving sufficient mask quality for qualitative inspection and consistency-based evaluation. Therefore, the final reported experiment used the `balanced` profile for both clean and corrupted inputs.
Alternative profiles were used only during pilot testing. The final results use the same `balanced` profile for clean and corrupted inputs so that observed differences can be attributed to corruption effects rather than inference-setting changes.

### 2.3.3 Mask Generation Procedure

For each input image, SAM2 generated a set of masks represented as dictionaries containing the binary segmentation and associated metadata. These masks were saved as compressed `.npz` files. The raw `.npz` files were used for all quantitative evaluation, because they preserve the actual mask outputs produced by SAM2.
In addition to the raw mask files, the inference pipeline generated qualitative overlay images. These overlays were created by sorting masks by area and painting them onto the image to produce a dense coloured visualisation. The overlays were used only for visual inspection and report figures; they were not used for the computation of IoU or other metrics. This distinction was important because the overlay is a visual aid, while the raw masks are the actual data used for evaluation.

### 2.3.4 Resumable Inference and Logging

SAM2 automatic mask generation was computationally expensive, especially when repeated across several corruption types and severity levels. To make the experiment manageable, the inference script was designed to support resumable execution. If a valid mask file and visualisation already existed for an image, the script skipped that image rather than processing it again. If the raw mask file existed but the visualisation was missing, the script regenerated only the visual output. This made it possible to run the experiment incrementally, pause and

resume processing, and add new images without restarting the entire pipeline.
The script also recorded per-image metadata in structured log files. These logs included the input filename, corruption type, severity level, image resolution, profile used, checkpoint path, runtime, number of generated masks, processing status, and output paths. This logging was useful for reproducibility and for checking whether corrupted images affected not only the segmentation masks but also practical behaviour such as runtime and mask count.
The output directory was organised by split, corruption type, and severity level. For example, clean-image masks were saved under the clean output folder, while corrupted-image masks were saved under folders such as `rain/level_1`, `fog/level_3`, and `snow/level_5`. This structure made the later consistency evaluation and qualitative grid generation simpler and reduced the risk of accidentally comparing mismatched images.

## 2.4 Evaluation Metrics

### 2.4.1 Rationale for Clean-vs-Corrupted Consistency

The primary evaluation in this project uses a clean-vs-corrupted consistency protocol. Instead of treating the Cityscapes ground truth as the main reference for every comparison, SAM2's output on the clean image is used as the reference and is compared with SAM2's output on the corrupted version of the same image.
This design was chosen because SAM2 is a class-agnostic foundation segmentation model rather than a conventional semantic segmentation network. It produces sets of masks, but these masks are not assigned semantic labels such as road, car, person, or building [15]. Therefore, a direct semantic mIoU-style evaluation is not the most natural primary metric for this project. The aim is instead to measure whether SAM2 remains stable when the appearance of the same underlying scene is changed by weather corruption.
The clean image provides a natural baseline because it represents SAM2's segmentation of the original uncorrupted scene. If the model is robust to a given corruption, the masks generated from the corrupted image should cover broadly similar foreground regions to those generated from the clean image. If the corruption disrupts SAM2's perception, the foreground structure, mask coverage, and number of masks are expected to change. This makes clean-vs-corrupted consistency suitable for measuring output stability under adverse visual conditions.

### 2.4.2 Dense Foreground Map Construction

SAM2 automatic mask generation produces an unordered set of masks for each image. Since the masks do not have fixed semantic labels or stable instance IDs across clean and corrupted images, comparing "mask 1" in the clean image with "mask 1" in the corrupted image would not be meaningful. The ordering of masks can change between runs and between image conditions. To avoid this problem, each set of SAM2 masks was converted into a dense foreground representation before computing the main overlap metric. Let the SAM2 output for an image be a set of binary masks:

$$\mathcal{M} = \{m_1, m_2, \ldots, m_n\}, \tag{2.1}$$

where each mask $m_i(x)$ indicates whether pixel $x$ belongs to that predicted region. Since SAM2 masks can overlap, the masks were first sorted by area in descending order. A dense label map $D(x)$ was then created by assigning each pixel to the first mask that covered it in this area-sorted order:

$$D(x) = \begin{cases} k, & \text{where } k = \min\{i : m_{\pi_i}(x) = 1\}, \\ 0, & \text{if no SAM2 mask covers pixel } x. \end{cases} \tag{2.2}$$

Here, $\pi$ denotes the ordering of masks from largest to smallest area. The value 0 represents pixels that were not assigned to any SAM2 mask.
The dense label map was then converted into a binary foreground map:

$$F(x) = \begin{cases} 1, & \text{if } D(x) > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{2.3}$$

This foreground map ignores arbitrary mask identities and instead measures whether SAM2 assigns each pixel to any predicted region. This is more appropriate for a class-agnostic automatic mask generator because it focuses on segmentation coverage and stability rather than semantic class prediction.

### 2.4.3 Foreground Consistency IoU

For each image and corruption condition, the foreground map from the clean image was compared with the foreground map from the corrupted image. Let $F_{\text{clean}}$ denote the binary foreground map produced from the clean-image SAM2 masks, and let $F_{\text{corr}}$ denote the corresponding foreground map from the corrupted image. The foreground consistency Intersection-over-Union (IoU) is defined as:

$$\text{IoU}(F_{\text{clean}}, F_{\text{corr}}) = \frac{|F_{\text{clean}} \cap F_{\text{corr}}|}{|F_{\text{clean}} \cup F_{\text{corr}}|}. \tag{2.4}$$

A higher foreground consistency IoU indicates that SAM2 produced a similar foreground segmentation structure on the corrupted image as it did on the clean image. A lower value indicates that the corruption caused a stronger change in the segmentation output. Therefore, decreasing foreground consistency as severity increases is interpreted as reduced robustness. This metric was selected as the main quantitative measure because it avoids direct comparison of arbitrary mask IDs. It also avoids relying on ground-truth-derived prompts or semantic class mappings, both of which would be less aligned with the automatic and class-agnostic nature of SAM2.

### 2.4.4 Supporting Consistency Metrics

In addition to foreground consistency IoU, several supporting metrics were computed to provide a broader view of robustness behaviour.
The Dice score was used as an alternative overlap measure:

$$\text{Dice}(F_{\text{clean}}, F_{\text{corr}}) = \frac{2|F_{\text{clean}} \cap F_{\text{corr}}|}{|F_{\text{clean}}| + |F_{\text{corr}}|}. \tag{2.5}$$

The clean foreground preserved ratio measured how much of the clean-image foreground remained covered under corruption:

$$\text{Preserved Ratio} = \frac{|F_{\text{clean}} \cap F_{\text{corr}}|}{|F_{\text{clean}}|}. \tag{2.6}$$

An instance-level best-match mask IoU was also computed. For each clean-image mask, the corrupted-image mask with the highest IoU was identified, and these best-match scores were averaged:

$$\text{BestMatchIoU} = \frac{1}{|\mathcal{M}_{\text{clean}}|} \sum_{m_i \in \mathcal{M}_{\text{clean}}} \max_{m_j \in \mathcal{M}_{\text{corr}}} \text{IoU}(m_i, m_j). \tag{2.7}$$

This provided a more instance-level view of whether individual clean-image masks had corresponding masks under corruption. Finally, mask count difference was recorded as:

$$\Delta N = |\mathcal{M}_{\text{corr}}| - |\mathcal{M}_{\text{clean}}|. \tag{2.8}$$

This helped identify whether weather corruption caused SAM2 to produce fewer or more masks than it did on the clean image.

### 2.4.5 Supplementary GT-Based Sanity Check

Although foreground consistency was the primary evaluation metric, a supplementary GT-based sanity check was also performed using Cityscapes label masks [3]. This was included to provide additional evidence that adverse weather corruptions reduce segmentation overlap, while keeping the main analysis centred on clean-vs-corrupted consistency.
For this check, each valid ground-truth semantic label region present in the Cityscapes label image was compared with the best-overlapping SAM2 mask. Background and void labels were excluded. This produces a lightweight best-overlap score rather than a full semantic segmentation mIoU, because SAM2 automatic masks do not contain semantic class labels.
Let $G_c$ denote the binary region for a valid ground-truth label $c$, and let $\mathcal{M}$ be the set of SAM2 masks for the corresponding image. For each ground-truth label region, the best overlap score was computed as:

$$s_c = \max_{m_j \in \mathcal{M}} \text{IoU}(G_c, m_j). \tag{2.9}$$

The final GT-based overlap score for an image was then computed by averaging across the valid labels present in that image:

$$S_{\text{GT}} = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} s_c, \tag{2.10}$$

where $\mathcal{C}$ is the set of valid ground-truth labels used in the comparison.
No ground-truth information was used to generate SAM2 masks. The GT labels were used only after inference as a post-hoc supplementary overlap check. Therefore, this score should not be interpreted as a full semantic segmentation benchmark. Instead, it is treated as a lightweight sanity check: if clean images achieve higher overlap scores than corrupted images, this supports

the broader conclusion that adverse weather degrades segmentation quality. This supplementary analysis is similar in spirit to prior work evaluating SAM under adverse-weather conditions, but it remains secondary to the foreground consistency analysis used in this project [17].

### 2.4.6 Evaluation Protocol

For each of the five selected Cityscapes images, SAM2 outputs from the clean image were compared against outputs from five corruption types at three severity levels. This produced:

$$5 \times 5 \times 3 = 75 \tag{2.11}$$

clean-corrupted evaluation pairs.
The final analysis grouped results by corruption type and severity level. Mean and standard deviation were computed across the five images for each condition. This allowed the report to examine both the average robustness trend and the amount of scene-level variation.
In cases where corrupted images were generated at a different spatial resolution from the original Cityscapes clean images, the corrupted masks were resized to match the clean-mask dimensions before computing overlap metrics. Nearest-neighbour interpolation was used because the masks are discrete binary or label maps, and nearest-neighbour resizing preserves hard mask boundaries without introducing fractional labels.
The per-image results were then aggregated by corruption type and severity level, and the analysis script generated summary tables, line plots, error-bar plots, severity-level bar charts, and supporting-metric plots.

### 2.4.7 Evaluation Design Choices

The final evaluation design balanced methodological rigour with practical feasibility. Five Cityscapes images were used for the quantitative subset, giving 75 clean-corrupted evaluation pairs across five corruption types and three severity levels. This scale was sufficient for a controlled small-scale analysis while remaining manageable for image generation, SAM2 inference, manual checking, and report preparation. A larger 10–20 image study would provide stronger statistical evidence, but was outside the available time and compute budget.
The experiment used three representative severity levels: Level 1, Level 3, and Level 5. These correspond to light, moderate, and severe corruption, allowing severity trends to be examined without running all five possible levels used in some corruption-robustness benchmarks. This reduces direct comparability with five-level studies, but keeps the experiment tractable while still capturing the main effect of increasing degradation.
Foreground consistency IoU was used as the primary metric because SAM2 produces class-agnostic, unordered masks rather than semantic class predictions. The clean SAM2 output therefore provides a natural reference for measuring how much the corrupted output changes. The GT-based overlap analysis was kept as a supplementary sanity check because Cityscapes labels are semantic, while SAM2 automatic masks do not directly provide semantic class labels. Qualitative figures were included alongside quantitative metrics because numerical scores alone do not show how the masks change. Corruption grids and SAM2 overlay grids help interpret

visible failure modes, including reduced foreground coverage, fragmentation, loss of small structures, and scene-dependent behaviour such as the non-monotonic rain results.

## 2.5 Implementation Details

### 2.5.1 Code Structure

The final implementation was organised into separate scripts for inference, evaluation, analysis, and qualitative visualisation:

- `run_sam2.py`: SAM2 inference on clean and corrupted images
- `sam_eval_utils.py`: shared evaluation utilities
- `compute_consistency.py`: primary clean-vs-corrupted consistency evaluation
- `analyze_consistency.py`: plotting and summary generation
- `make_qual_grids.py`: generation of qualitative comparison grids

A secondary GT-based evaluation path was retained in `compute_iou.py`, but this was treated as supplementary rather than the main experimental route.

### 2.5.2 Validation and Testing

The final codebase was validated through a combination of metric-level sanity tests, file-structure checks, and end-to-end pipeline verification. Since the main evaluation depends on foreground consistency IoU, the overlap metric implementation was checked using simple synthetic masks with known expected results.
A dedicated validation script, `test_sam_eval_utils.py`, was added to test the shared metric utilities. These tests confirmed that the IoU function returns 1.0 for identical masks, 0.0 for non-overlapping masks, and the expected value for partially overlapping masks. The Dice score was similarly checked on known binary masks. The nearest-neighbour resizing utility was also tested to confirm that discrete mask labels are preserved and that interpolation does not introduce invalid intermediate values.
The implementation was also checked for portability and clarity. The representative qualitative-grid script was refactored to remove hardcoded absolute Windows paths and now uses command-line arguments for the run directory, image name, and output directory. Legacy exploratory scripts were separated from the final pipeline to avoid confusion between early development work and the final reported experiment.

Table 2.3: Implementation validation checks used in the final project.

| Component | Validation check | Purpose |
|---|---|---|
| Binary IoU | Tested on identical, non-overlapping, and partially overlapping synthetic masks | Confirms that the primary overlap metric returns known expected values. |
| Dice score | Tested on simple binary masks | Confirms correctness of the supporting overlap metric. |
| Nearest-neighbour resizing | Tested on discrete labelled masks | Ensures mask labels remain discrete after resizing. |
| Filename mapping | Clean, corrupted, mask, and GT files matched by shared base filename | Reduces risk of comparing mismatched scenes. |
| Resumable inference | Re-running completed cases skips valid existing outputs | Confirms that long SAM2 runs can be resumed safely. |
| Script portability | Removed hardcoded absolute paths from qualitative-grid generation | Improves reproducibility on another machine. |
| Final/legacy separation | Legacy scripts moved to an exploratory folder | Makes clear which scripts produced the final reported results. |

## 2.6 Version Control and Artefact Management

The final project artefacts were organised in a separate folder named `SAM2_Robustness_Project`. A public-safe GitHub repository was also prepared to provide access to the final code, validation tests, result summaries, quantitative plots, README documentation, and reproducibility notes.

To avoid licensing and file-size issues, the public repository does not include raw Cityscapes images, generated Nano Banana Pro corrupted images, Cityscapes ground-truth label files, raw SAM2 `.npz` mask outputs, qualitative overlay images, or SAM2 checkpoint files. These full artefacts are instead organised in the complete project folder and provided separately via a OneDrive link by email for assessment access.

Large SAM2 checkpoint files were excluded from the repository because of file size. The checkpoint used was `sam2.1_hiera_large.pt`, with configuration `sam2.1/sam2.1_hiera_l.yaml`. This separation keeps the public repository suitable for inspection of the implementation while keeping restricted or large artefacts within the approved submission route.

# Chapter 3

# Experiments and Results

## 3.1 Experimental Setup

### 3.1.1 Conditions

The final experiments used one clean baseline and five visual corruption conditions: rain, fog, snow, frost, and brightness. Each corruption was evaluated at three severity levels: Level 1, Level 3, and Level 5. Combined with the five selected images, this produced a total of 75 clean-corrupted evaluation pairs.

### 3.1.2 Compute Resources

Experiments were run locally on a laptop GPU environment using a dedicated conda setup for SAM2 inference; the full hardware and software configuration is listed in Table B.2. Runtime was an important practical constraint throughout the project, particularly for automatic mask generation and for consistency analysis involving large numbers of mask comparisons. To keep the final evaluation tractable, the inference pipeline supported resumable execution and filtering by image, corruption type, and severity level. This made it possible to process images incrementally while maintaining a consistent final experimental configuration.

## 3.2 Qualitative Results

### 3.2.1 Corruption Grids

Visual inspection of the corrupted images confirmed that the selected corruption types produced distinct degradation patterns. Fog reduced global visibility and contrast, brightness corruption washed out image details, and snow, rain, and frost introduced structured artefacts that partially obscured scene content. Across the three chosen severity levels, frost and heavy snow produced the most visually disruptive changes, particularly in regions containing fine structure and texture. This qualitative impression is consistent with the quantitative results in Table 3.1, where frost and snow show the strongest high-severity degradation.
These qualitative grids are useful because they show that the corruption pipeline does not simply darken or brighten the image uniformly; rather, each corruption changes the visual evidence available to SAM2 in a different way. The final reported corruption images were generated using Nano Banana Pro, which produced more realistic visual effects than the earlier augmentation-based exploratory pipeline [6, 1].

### 3.2.2 Segmentation Visualisations

The qualitative SAM2 overlays showed several recurring failure modes under heavier corruption:

- reduced segmentation coverage in low-contrast regions;

- fragmentation of larger scene regions into smaller masks;
- loss of thin or small structures;
- substantial changes in total mask count between clean and corrupted inputs.

These effects were most visible under strong frost and snow, where the final overlays diverged substantially from the clean-image outputs. Fog and brightness caused milder changes, while rain produced more variable behaviour across images.
The most affected scene elements appeared to be fine or low-contrast structures, distant objects, object boundaries, and textured road regions. Large high-contrast regions were often more stable, while smaller structures and partially occluded areas were more likely to be missed, merged, or fragmented. This interpretation is qualitative because the primary SAM2 outputs are class-agnostic masks rather than semantic class predictions.




Figure 3.1: Representative corruption grid for `frankfurt_000001_064925_leftImg8bit`. Rows show the clean image and three corruption severity levels, while columns correspond to the five visual corruption types used in this study.

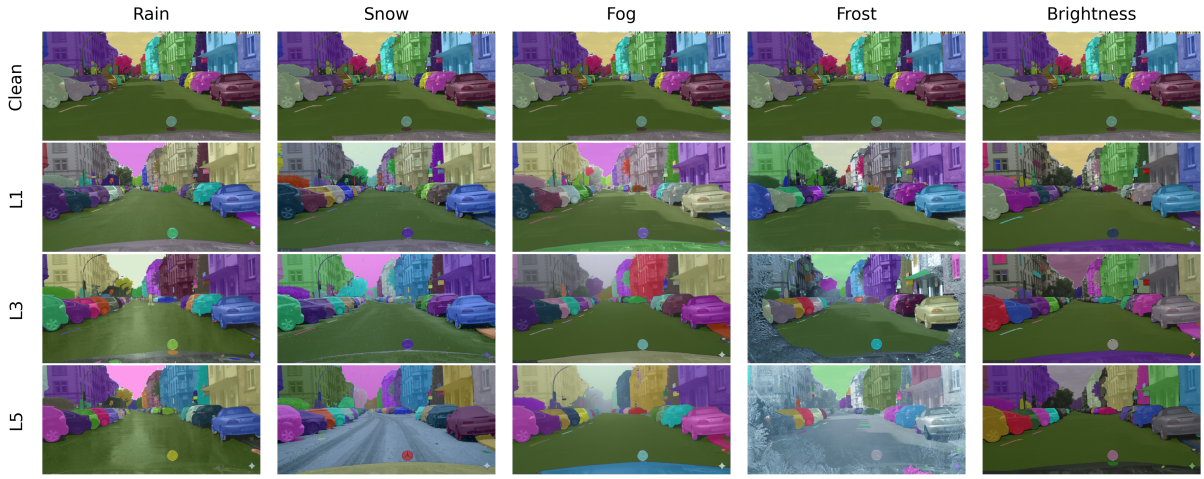



Figure 3.2: Representative SAM2 segmentation outputs for `frankfurt_000001_064925_leftImg8bit`, matching the same layout as Figure 3.1. This allows direct visual comparison of how SAM2 output structure changes as corruption severity increases.

## 3.3 Quantitative Results

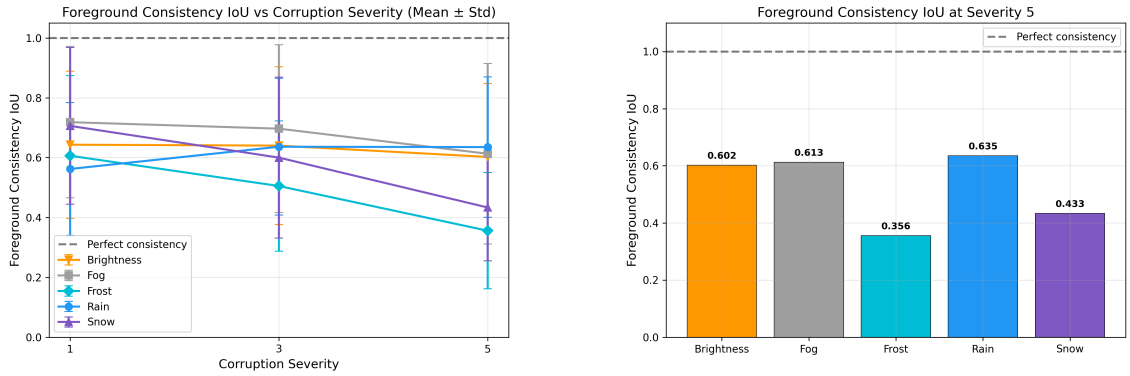
### 3.3.1 Foreground Consistency IoU Across Severity

The main quantitative metric was foreground consistency IoU between clean and corrupted SAM2 outputs. Table 3.1 reports the mean and standard deviation across the five selected images for each corruption type and severity level. Figure 3.3a visualises the same trend with standard-deviation error bars.

Table 3.1: Primary foreground consistency IoU results for the final five-image evaluation. Values show mean ± standard deviation across five selected Cityscapes images.

| **Condition** | **Level 1** | **Level 3** | **Level 5** |
|---|---|---|---|
| Brightness | $0.6431 \pm 0.2461$ | $0.6401 \pm 0.2639$ | $0.6022 \pm 0.2452$ |
| Fog | $0.7184 \pm 0.2530$ | $0.6967 \pm 0.2802$ | $0.6128 \pm 0.3014$ |
| Frost | $0.6062 \pm 0.2678$ | $0.5051 \pm 0.2179$ | $0.3555 \pm 0.1942$ |
| Rain | $0.5620 \pm 0.2219$ | $0.6363 \pm 0.2287$ | $0.6351 \pm 0.2346$ |
| Snow | $0.7060 \pm 0.2621$ | $0.6000 \pm 0.2691$ | $0.4332 \pm 0.1780$ |

Overall, most corruption types showed reduced consistency as severity increased. The clearest monotonic trend was observed for frost, where mean foreground IoU decreased from 0.6062 at Level 1 to 0.5051 at Level 3 and 0.3555 at Level 5. Snow also showed a substantial drop at the highest severity, decreasing from 0.7060 at Level 1 to 0.4332 at Level 5. In contrast, fog and brightness produced more moderate reductions, with mean IoU remaining above 0.60 at Level 5. Rain was the only corruption that did not follow a monotonic trend: its mean IoU increased from 0.5620 at Level 1 to 0.6363 at Level 3 and remained similar at 0.6351 at Level 5. This suggests that, in this small-scale evaluation, the relationship between rain severity and SAM2 consistency was more image-dependent than for the other corruption types. This should not be interpreted as evidence that heavier rain improves SAM2 robustness. A more cautious interpretation is that rain effects were more scene-dependent in the small five-image subset. In particular, light rain can introduce thin streaks and local boundary-like artefacts without substantially changing the global scene appearance, while heavier rain may produce more uniform haze or wet-road effects that affect larger regions more consistently. With only five scenes, individual image behaviour can therefore influence the mean trend strongly. The rain result is best treated as an observed non-monotonic pattern that motivates further investigation on a larger dataset.




(a) Foreground consistency IoU across corruption types and severity levels. Error bars show standard deviation across the five evaluated images.

(b) Mean foreground consistency IoU at the highest evaluated severity level (Level 5). Lower values indicate greater robustness degradation.

Figure 3.3: Quantitative robustness results for the five-image evaluation subset. The left plot shows mean foreground consistency IoU across severity levels for each corruption type, while the right plot summarises the Level 5 comparison directly.

The same trend is also visible when the results are expressed as consistency drop relative to the clean output, shown in Appendix B.1.

### 3.3.2 Comparison Across Corruption Types

To compare corruption types directly under the strongest setting, Figure 3.3b shows mean foreground consistency IoU at Level 5. The bars in Figure 3.3b are ordered alphabetically by corruption type, while the discussion below describes them from lowest to highest robustness. Frost was the most damaging corruption by a clear margin, with mean IoU dropping to 0.3555. Snow was the next most damaging at 0.4332. Brightness, fog, and rain retained higher consistency at Level 5, with values of 0.6022, 0.6128, and 0.6351 respectively.
This ranking indicates that structured occlusion-like effects and strong texture changes can be more disruptive to SAM2's segmentation consistency than global visibility or exposure changes.

### 3.3.3 Scene Variability

Although the mean trends are informative, several standard deviations in Table 3.1 are large relative to their corresponding means, for example fog at Level 5 ($0.6128 \pm 0.3014$) and snow at Level 3 ($0.6000 \pm 0.2691$). This indicates variation across the five images and suggests that robustness is strongly scene-dependent. In other words, the effect of a given corruption does not appear uniform across all urban scenes.
This observation is consistent with the qualitative outputs: some scenes retained relatively stable foreground structure under corruption, while others exhibited stronger mask loss or fragmentation. As a result, the reported averages should be interpreted as indicative trends within a small controlled subset rather than definitive dataset-wide behaviour.

Table 3.2: Paired change in foreground consistency IoU from Level 1 to Level 5. Positive values indicate lower consistency at Level 5.

| Condition | Mean L1–L5 drop | Std. of drop | Images degraded |
|---|---|---|---|
| Brightness | 0.0409 | 0.0439 | 4/5 |
| Fog | 0.1056 | 0.1802 | 4/5 |
| Frost | 0.2507 | 0.2830 | 5/5 |
| Rain | -0.0731 | 0.1969 | 2/5 |
| Snow | 0.2729 | 0.1974 | 4/5 |

Table 3.2 provides an additional paired view of the severity effect. Frost and snow show the largest average drops from Level 1 to Level 5, while rain shows a negative mean drop because the Level 5 outputs were not consistently worse than Level 1 across the five scenes. For rain, only two of the five scenes degraded from Level 1 to Level 5, meaning that the majority of scenes showed equal or improved foreground consistency at Level 5 relative to Level 1. Given the small sample size, these values should be interpreted as descriptive evidence rather than formal statistical proof.

### 3.3.4 Supporting Metrics

Additional metrics provided supporting evidence for the main foreground IoU results. Table 3.3 summarises selected supporting metrics at the highest severity level.

Table 3.3: Selected supporting metrics at Level 5. Values are means across the five selected images.

| Condition | Fg IoU | Dice | Best-match IoU | Mask count Δ |
|---|---|---|---|---|
| Brightness | 0.6022 | 0.7186 | 0.2865 | -159.6 |
| Fog | 0.6128 | 0.7099 | 0.3097 | -122.4 |
| Frost | 0.3555 | 0.4948 | 0.2349 | -175.8 |
| Rain | 0.6351 | 0.7507 | 0.2927 | -107.6 |
| Snow | 0.4332 | 0.5816 | 0.2342 | -134.4 |

Frost and snow again show weaker results in both foreground IoU and Dice score. They also have low best-match mask IoU, suggesting that individual clean-image masks often lacked strong corresponding masks under severe corruption. The negative mask-count differences show that corrupted images generally produced fewer masks than their clean counterparts, with the largest reductions observed for frost and brightness.

These supporting metrics reinforce the main conclusion that adverse corruptions can alter not only the spatial extent of SAM2's foreground predictions but also the granularity of its mask decomposition.

While the previous sections focus on the primary clean-vs-corrupted consistency evaluation, the following subsection reports the supplementary GT-based check requested to provide a second form of evidence.

### 3.3.5 Supplementary GT-Based Sanity Check

A lightweight GT-based sanity check was also performed using the same five-image subset. This analysis compared each Cityscapes ground-truth semantic region to the best-overlapping SAM2 mask and averaged these overlaps across the labels present in each image. The clean baseline score was 0.3264, while all corrupted condition-level averages were lower than the clean case. This should not be compared directly with conventional semantic segmentation mIoU or with Shan and Zhang's SAM results, because the score is a best-overlap check over class-agnostic SAM2 masks rather than class-labelled semantic prediction.

Table 3.4: Supplementary GT-based sanity-check results (mean overlap score across five images).

| Condition | Level 1 | Level 3 | Level 5 |
|---|---|---|---|
| Brightness | 0.2454 | 0.2410 | 0.2124 |
| Fog | 0.2452 | 0.2436 | 0.2147 |
| Frost | 0.2089 | 0.1890 | 0.1575 |
| Rain | 0.2072 | 0.2179 | 0.1890 |
| Snow | 0.2398 | 0.2266 | 0.2174 |
| Clean baseline | | 0.3264 | |

The clean GT-based sanity-check score was 0.3264, which is much lower than the values usually expected from a supervised semantic segmentation model on clean Cityscapes. This is expected because the score is not a conventional semantic mIoU. SAM2 automatic mask generation produces class-agnostic masks without semantic labels, while the Cityscapes annotations are semantic regions. The GT-based check therefore measures whether each semantic region has a reasonably overlapping SAM2 mask, not whether SAM2 assigns the correct class label. The low clean baseline indicates that SAM2 automatic masks do not naturally align perfectly with Cityscapes semantic categories, even before weather corruption is applied. For this reason, the GT-based result is used only as supplementary evidence rather than as the main evaluation. The GT-based summary broadly supported the main consistency analysis. Frost again showed the strongest degradation, decreasing from 0.2089 at Level 1 to 0.1575 at Level 5. Snow, fog, and brightness also declined under stronger corruption, while rain again displayed a non-monotonic pattern, rising slightly at Level 3 before dropping at Level 5. Although this supplementary analysis is not the main evaluation route, it provides additional evidence that adverse weather generally reduces segmentation overlap, consistent with prior adverse-weather SAM analysis [17].

# Chapter 4

# Discussion, Limitations, and Conclusion

## 4.1 Summary of Findings

The quantitative results support four main findings:

1. SAM2 consistency generally decreases as corruption severity increases, but the trend is not perfectly monotonic for every corruption type.
2. Frost is the most damaging corruption in this controlled evaluation, with severe snow also causing substantial degradation.
3. The relationship between corruption severity and robustness is scene-dependent, as shown most clearly by the rain results and the large standard deviations across some conditions.
4. A supplementary GT-based sanity check shows the same broad direction of degradation, with clean performance higher than all corrupted conditions.

Taken together, these findings suggest that SAM2 is sensitive to adverse weather-related visual degradation, but that the magnitude and smoothness of this degradation depend strongly on the corruption type and the content of the scene.

## 4.2 Discussion

The results show that SAM2 segmentation consistency generally decreases as corruption severity increases, although the strength and smoothness of this effect depend on the corruption type. Across the evaluated conditions, frost produced the strongest degradation, with a particularly sharp drop at the highest severity. Snow also caused a substantial decline at Level 5, while fog and brightness produced more moderate reductions. Rain was the only corruption that did not follow a monotonic trend across the three evaluated levels.
These findings suggest that SAM2 is not affected equally by all forms of visual degradation. Corruptions such as frost and snow introduce strong structured artefacts or occlusion-like effects that appear to disrupt SAM2's ability to preserve the same segmentation structure seen in the clean image. By contrast, brightness and fog mainly alter visibility and contrast, which in this small-scale study appeared less destructive to the overall foreground layout.
The qualitative overlays help explain these trends. Under stronger corruptions, the model frequently produced reduced foreground coverage, fragmented regions, or fewer overall masks than in the clean case. This is consistent with the supporting mask-count analysis, which showed that corrupted images often yielded substantially fewer masks than their clean counterparts. In other words, degradation was not only reflected in overlap scores, but also in the granularity of SAM2's decomposition of the scene.
An important result is that the effect of corruption remained scene-dependent even after expanding the evaluation from three to five images. The standard deviations in several plots

indicate that robustness varied across the selected scenes, showing that the same corruption type did not affect every image equally. Certain scenes retained relatively stable segmentation under moderate corruption, while others changed much more dramatically. As a result, the project's conclusions should still be interpreted as controlled observations on a small selected subset rather than definitive statements about all urban scenes.
The non-monotonic trend observed for rain is also important. Rather than forcing a simple narrative that all corruptions worsen smoothly with severity, the results suggest that some corruption types can interact with scene content in more complex ways. It is plausible that, for some images, a medium-level rain pattern preserved enough structural cues to remain closer to the clean segmentation than the lighter or heavier settings. This highlights the need for caution when interpreting robustness curves from small samples.

## 4.3 Answers to the Research Questions

### RQ1: How does corruption severity affect SAM2 segmentation consistency?

The results show that SAM2 segmentation consistency generally decreases as corruption severity increases, although the trend is not perfectly monotonic for every corruption type. Frost and snow showed the clearest degradation from Level 1 to Level 5, while brightness and fog showed more moderate reductions. Rain was more scene-dependent and did not show a consistent monotonic decrease.

### RQ2: Which weather corruptions are most damaging?

At Level 5, frost produced the lowest mean foreground consistency IoU, followed by snow. This indicates that structured texture changes and occlusion-like effects were more disruptive in this controlled evaluation than brightness, fog, or rain. The supplementary GT-based sanity check supported the same broad conclusion, with frost again producing the lowest overlap score.

### RQ3: What failure modes are visible in the SAM2 outputs?

The qualitative overlays suggest several recurring failure modes: reduced foreground coverage in low-contrast regions, loss of small or thin structures, fragmentation of larger regions, and changes in mask granularity. These effects appeared most strongly under frost and snow. Fine structures, distant objects, object boundaries, textured road regions, and smaller foreground objects appeared more vulnerable than large high-contrast regions. However, this observation is qualitative because SAM2 automatic masks are class-agnostic and do not provide semantic class labels.

## 4.4 Methodological Reflections

As discussed in Section 2.4.7, the main methodological decision was to use clean-image SAM2 outputs as the reference for corrupted outputs. This matched the class-agnostic nature of SAM2 and allowed the project to measure output stability rather than semantic accuracy. The

supplementary GT-based sanity check was therefore used only as supporting evidence, not as the main benchmark.

A second important methodological decision was to move away from relying on conventional augmentation tools such as Albumentations [1] for the final corruption set. Although such tools are efficient and reproducible, they often produce visually simpler effects than higher-quality image-to-image generation. The final reported experiments therefore used Nano Banana Pro to generate more realistic corrupted images, while the earlier augmentation-based pipeline was retained only as a legacy or exploratory component of the codebase.

A third important methodological issue was output resolution mismatch. Some corrupted images were available at a larger spatial resolution than the original clean Cityscapes images, so the evaluation pipeline resized corrupted foreground masks to the clean-image dimensions using nearest-neighbour interpolation before computing overlap. This was a practical and appropriate solution, since nearest-neighbour preserves discrete mask boundaries, but it remains an additional methodological step that should be acknowledged.

Finally, the supplementary GT-based sanity check was useful because it supported the same broad direction as the main consistency results without replacing them. In both analyses, clean performance was higher than corrupted performance, frost emerged as the most damaging corruption, and rain remained less stable as a severity trend. This strengthened confidence that the consistency-based findings were not artefacts of a single metric alone.

## 4.5 Limitations

This project has several limitations.

First, the final controlled evaluation used five selected Cityscapes images. This was stronger than the earlier three-image pilot and produced 75 clean-corrupted evaluation pairs, but it is still small compared with a benchmark-scale robustness study. The reported trends should therefore be interpreted as controlled observations rather than definitive dataset-wide conclusions.

Second, the corruptions used in the study were synthetic rather than captured from real adverse-weather driving scenes. Although this made the experiment controllable, synthetic corruptions cannot fully reproduce the complexity of real weather, camera optics, road surface conditions, sensor behaviour, and lighting variation. The interpretation of per-corruption rankings is also limited by the small sample size. Some standard deviations are large relative to the corresponding means, which means that the results should be read as indicative patterns rather than statistically robust rankings. This is particularly relevant for rain, where the mean trend was non-monotonic and likely influenced by scene-specific behaviour.

Third, the final corruption-generation workflow used Nano Banana Pro through the Gemini web/app interface. This improved visual realism compared with the early augmentation-based script, but it reduced reproducibility compared with a deterministic augmentation pipeline. The final corruptions were not generated through a scripted API call with fixed seeds and versioned parameters. Although prompt constraints and manual checks were used to preserve scene geometry, another researcher may not be able to regenerate identical corrupted images from the same prompts.

Fourth, the scene-preservation check was practical rather than fully objective. The generated images were manually inspected to check that major scene geometry, camera viewpoint, vehicles, pedestrians, and road layout were preserved. However, no formal inter-rater annotation, SSIM/PSNR threshold, or feature-alignment metric was used. This means the corruption quality-control process contains a subjective component.
Fifth, SAM2 produces class-agnostic and unordered masks. The clean-vs-corrupted foreground consistency approach is suitable for measuring output stability, but it does not directly measure semantic correctness. A high consistency score means that the corrupted output remains similar to the clean SAM2 output; it does not guarantee that either output is semantically correct.
Finally, the project was constrained by runtime and computation. High-quality SAM2 inference settings were too slow for systematic use across all conditions, so the final experiment used the balanced profile. This made the study feasible, but future work could repeat the same evaluation with heavier inference settings or additional model variants.

## 4.6 Methodological Lessons

The project also highlighted several methodological lessons that are relevant to future robustness studies using foundation segmentation models. First, directory structure and naming consistency are not minor implementation details in this type of experiment. Because each clean image must be matched with corrupted images, SAM2 masks, qualitative overlays, and ground-truth files, deterministic filename mapping was essential for avoiding accidental comparison errors.
Second, the use of a fixed SAM2 inference profile was important for fair comparison. If clean and corrupted inputs were processed using different automatic mask-generation settings, changes in the results could not be attributed confidently to the corruptions themselves. Keeping the inference profile constant therefore made the clean-versus-corrupted comparison more meaningful.
Third, qualitative evidence was necessary alongside quantitative scores. Foreground consistency IoU summarises output stability, but it does not explain why masks change. Corruption grids and SAM2 overlay figures helped identify visible failure patterns such as texture disruption, reduced boundary clarity, and changes in mask granularity.
Finally, the corruption-generation stage showed a trade-off between realism and reproducibility. The final image-to-image workflow produced more plausible visual corruptions than the early classical augmentation script, but it was less deterministic and harder to scale. This trade-off is an important consideration for future work on robustness evaluation under synthetic adverse conditions.

## 4.7 Future Work

Several directions could extend this work.
The most immediate improvement would be to scale the evaluation to a larger and more diverse subset of Cityscapes images. This would reduce scene-specific variance and make the robustness trends more statistically reliable. With a larger sample, confidence intervals or

formal statistical tests could be used to distinguish stronger effects from noise.
A second extension would be to evaluate SAM2 on real adverse-weather driving datasets such as ACDC, which contains real fog, rain, snow, and nighttime driving scenes. This would test whether the trends observed in the synthetic corruption setting transfer to real adverse conditions.
A third improvement would be to use a more reproducible corruption-generation process. One option would be to return to deterministic augmentation, but with more realistic parameter tuning. Another option would be to archive the full prompt text, generated outputs, model version information, and quality-control metadata for each image-to-image generation.
A fourth extension would be to compare SAM2 with other segmentation models, including conventional semantic segmentation models and the original SAM. This would help determine whether the observed degradation patterns are specific to SAM2 or reflect a broader weakness of segmentation models under adverse weather.
Finally, future work could analyse scene elements more explicitly. Since this project used class-agnostic SAM2 masks, the discussion of affected scene regions was mainly qualitative. A larger study could combine SAM2 masks with semantic labels to examine whether roads, vehicles, pedestrians, vegetation, sky, or distant objects are affected differently by each corruption type.

## 4.8 Conclusion

This project developed a controlled and documented framework for evaluating SAM2 robustness under synthetic adverse-weather and brightness-related corruptions on urban driving images. The final experiments used five selected Cityscapes images, four adverse-weather corruption types (rain, fog, snow, and frost), one photometric brightness distortion, and three representative severity levels. Clean-vs-corrupted foreground consistency IoU was used as the primary metric, with a lightweight GT-based overlap analysis included as supplementary evidence.
The results showed that SAM2 segmentation consistency generally declines as corruption severity increases, with frost and severe snow producing the strongest degradation in this study. Fog and brightness caused more moderate degradation, while rain showed a more variable and non-monotonic trend. The supplementary GT-based check supported the same broad direction by showing that the clean baseline outperformed all corrupted conditions and that frost again produced the weakest overlap.
The project also showed that evaluating SAM2 requires a protocol that matches its class-agnostic output structure. Since SAM2 produces unordered masks rather than semantic class predictions, clean-anchored consistency provided a more suitable primary measure than conventional semantic mIoU alone. Although the study is limited in scale, it provides a clear methodology, implementation framework, and set of artefacts for future robustness analysis of foundation segmentation models under challenging driving conditions.

# References


[1] A. Buslaev, A. Parinov, E. Khvedchenya, V. I. Iglovikov, and A. A. Kalinin. Albumentations: Fast and flexible image augmentations. *Information*, 11(2):125, 2020.

[2] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 801–818, 2018.

[3] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[4] draw.io Ltd. About draw.io. `https://www.drawio.com/about`, 2026. Accessed: 27 April 2026.

[5] Google AI for Developers. Gemini 3 pro image preview: Nano banana pro. `https://ai.google.dev/gemini-api/docs/models/gemini-3-pro-image-preview`, 2026. Accessed: 27 April 2026.

[6] Google AI for Developers. Gemini image generation documentation: Nano banana and nano banana pro. `https://ai.google.dev/gemini-api/docs/image-generation`, 2026. Accessed: 27 April 2026.

[7] Google Research. Google colaboratory frequently asked questions. `https://research.google.com/colaboratory/faq.html`, 2026. Accessed: 27 April 2026.

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017.

[9] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations (ICLR)*, 2019.

[10] C. Kamann and C. Rother. Benchmarking the robustness of semantic segmentation models with respect to common corruptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[11] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023.

[12] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015.

[13] Meta AI. Segment anything model 2 official github repository. `https://github.com/facebookresearch/sam2`, 2024. Accessed: 27 April 2026.

[14] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[15] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations (ICLR)*, 2025. Oral.

[16] C. Sakaridis, D. Dai, and L. Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[17] X. Shan and C. Zhang. Robustness of segment anything model (sam) for autonomous driving in adverse weather conditions. *arXiv preprint arXiv:2306.13290*, 2023.

[18] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[19] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890, 2017.

# Appendix A

# Self-appraisal

## A.1 Critical Self-Evaluation

At the start of this project, I aimed to investigate how robust a modern foundation segmentation model is when urban driving images are affected by adverse weather conditions. More specifically, my goals were to generate weather-corrupted versions of Cityscapes images, run SAM2 on clean and corrupted inputs, evaluate the effect of corruption severity on segmentation stability, and present both quantitative and qualitative evidence in a clear final report.

In large part, I achieved these goals. I built an end-to-end experimental pipeline for organising clean and corrupted images, running SAM2 automatic mask generation, saving raw mask outputs, producing qualitative overlays, and generating quantitative results. The final evaluation used five selected Cityscapes images, five visual corruption conditions, and three severity levels, producing 75 clean-corrupted evaluation pairs. I also implemented a clean-vs-corrupted foreground consistency evaluation, which was appropriate for SAM2 because it is a class-agnostic mask generator rather than a conventional semantic segmentation model. Following supervisor feedback, I added a supplementary GT-based sanity check to provide additional evidence that adverse weather reduces segmentation overlap.

However, the project also had important limitations. The final dataset size remained small, so the results should be interpreted as a controlled small-scale study rather than a full benchmark evaluation. A larger study across 10–20 or more images would provide stronger statistical evidence and reduce the effect of scene-specific variation. The corruption-generation process also had limitations. Early classical augmentation outputs were fast and reproducible, but visually unrealistic in severe cases. The final Nano Banana Pro image-to-image workflow improved realism, but was less reproducible than a deterministic scripted pipeline and sometimes required additional handling because of output-resolution differences.

The evaluation protocol also had limitations. Foreground consistency IoU is useful for measuring output stability, but it is not the same as semantic segmentation accuracy. It shows whether SAM2 produces similar foreground coverage under corruption, but it does not prove that the predicted regions correspond to correct semantic classes. The supplementary GT-based check helped provide additional context, but it remained a lightweight sanity check rather than a full semantic benchmark.

There were also implementation challenges. SAM2 automatic mask generation was computationally expensive, especially with high-quality settings and high-resolution images. I therefore had to refactor the pipeline to support resumable execution, image filtering, condition filtering, and separate output files for different evaluation runs. These changes improved the final implementation, but they also reduced the time available for larger-scale experimentation.

If I had more time, I would improve the project in five main ways. First, I would expand the evaluation to a larger and more diverse set of Cityscapes images. Second, I would develop a

more reproducible corruption-generation process that preserves image resolution and scene geometry while still producing realistic weather effects. Third, I would run a fuller GT-based analysis while keeping consistency as the main metric for class-agnostic SAM2 outputs. Fourth, I would include confidence intervals or statistical tests once more images were available. Finally, I would compare SAM2 against the original SAM and one or more conventional segmentation baselines.
Overall, the project achieved its central aim: it developed and evaluated a controlled pipeline for studying SAM2 robustness under adverse-weather and brightness-related corruptions. The results showed that SAM2 consistency generally decreases under stronger corruption, especially for frost and snow, while rain showed a more scene-dependent and non-monotonic pattern. At the same time, the project exposed limitations in data scale, corruption reproducibility, compute time, and evaluation design. These limitations do not invalidate the findings, but they should be considered when interpreting the results.

## A.2 Legal, Social, Ethical and Professional Issues

Although this project did not involve human participants, interviews, or the collection of personal data, it still raised legal, social, ethical, and professional considerations. These issues are lower risk than in many user-centred or data-sensitive projects, but they remain important because the work evaluates computer-vision robustness in a safety-relevant application domain.

### A.2.1 Legal Issues

The project used publicly available academic research resources rather than confidential or proprietary data. The Cityscapes dataset was used only for academic research and evaluation, and no attempt was made to identify individuals, vehicles, or locations beyond what is already present in the dataset in anonymised research form. External software, pretrained models, and libraries, including SAM2 and Python packages, were used under their respective licences and acknowledged in the report and codebase.
Care was taken not to treat Cityscapes-derived data or generated corrupted images as unrestricted public repository content, because dataset images and modified versions may be subject to redistribution restrictions. The GitHub repository is therefore intended for code, documentation, validation tests, quantitative result summaries, and non-restricted plots only. Full Cityscapes-derived artefacts, generated corrupted images, SAM2 masks, qualitative overlays, and large output files are provided separately through the submitted project artefact folder via OneDrive link by email, where permitted.

### A.2.2 Social Issues

The project is socially relevant because robustness failures in perception systems may matter in safety-critical domains such as autonomous driving. If segmentation becomes less stable under adverse weather, downstream perception and decision-making systems may also become less reliable. Although this dissertation does not deploy a real vehicle system, it contributes positively by highlighting the need to evaluate foundation segmentation models beyond clean benchmark conditions.

At the same time, there is a social responsibility not to overstate the findings. The final evaluation is small-scale and uses synthetic corruptions, so the results are presented as indicative rather than universal.

### A.2.3 Ethical Issues

The main ethical issue is accurate reporting. Synthetic corruptions are useful for controlled evaluation, but they cannot fully represent the complexity of real adverse weather, sensor noise, optics, or traffic conditions. It would therefore be misleading to claim that the results fully represent real deployment performance.
A second ethical issue is interpretation of the metric. The primary metric measures clean-vs-corrupted consistency, not full semantic correctness. A high consistency score means that SAM2 behaves similarly under corruption, but it does not guarantee that the output is semantically correct. The report therefore distinguishes clearly between the primary consistency evaluation and the supplementary GT-based sanity check.

### A.2.4 Professional Issues

Several professional practices were followed during the project. The implementation used a structured directory layout, consistent filenames, separated scripts for inference, evaluation, plotting, and qualitative visualisation, and a fixed SAM2 inference profile for both clean and corrupted inputs. This reduced the risk of accidental inconsistency.
The final artefacts were organised into a separate submission folder, while a public GitHub repository was prepared for code, documentation, validation tests, result summaries, and non-restricted quantitative plots. Legacy exploratory scripts were separated from final scripts to reduce confusion. This helped make the final implementation easier to inspect and maintain.
Overall, while this project posed relatively low direct ethical risk, it still required careful attention to lawful dataset use, accurate reporting, responsible interpretation of robustness results, and reproducible professional research practice.

# Appendix B

# External Materials

## B.1 Project Repository and Artefacts

The final project artefacts are organised into a structured folder named `SAM2_Robustness_Project`. This folder contains the final code, result CSV files, report figures, validation tests, selected data artefacts, SAM2 outputs, and reproducibility notes. The final report PDF remains the primary submitted document, while the artefact folder provides supporting material for inspection and reproducibility.
A public GitHub repository was prepared for the final code and non-restricted result artefacts:

`https://github.com/Sachin-1712/SAM2_Robustness_Project_UoL`

The public repository contains the final implementation scripts, validation tests, README documentation, result summaries, and quantitative plots. It does not include raw Cityscapes images, generated Nano Banana Pro corrupted images, Cityscapes ground-truth label files, raw SAM2 `.npz` masks, qualitative overlay images, or SAM2 checkpoint files, because these may be restricted by dataset licensing or file-size constraints.
The complete project artefact folder is provided separately via a OneDrive link by email for assessment access, where permitted. This full folder contains the supporting files needed to inspect the final experimental outputs, including selected data artefacts, generated corruptions, SAM2 outputs, qualitative visualisations, result CSV files, logs, and report assets.
The final cleanup and validation commit for the public-safe repository was:

`524afa82a94218cebf65e05ed2f0e4f7999bb9ca`

## B.2 Submitted Artefact Structure

The supporting artefact folder follows this structure:

- `00_Final_Report/`: final report PDF and report-related notes.
- `01_Code/`: final scripts and legacy exploratory scripts.
- `02_Data_Subset/`: selected clean and corrupted image subset, where permitted.
- `03_SAM2_Outputs/`: raw SAM2 mask outputs, qualitative overlays, logs, and manifest files.
- `04_Results/`: consistency metrics, summary files, GT sanity-check outputs, and plots.
- `05_Report_Assets/`: report-ready figures.
- `06_Appendix_Extra_Figures/`: additional qualitative and supporting figures.

- `07_Checkpoints_Info/`: checkpoint and configuration notes.

This complete artefact folder is supplied separately via OneDrive link by email for assessment access, while the public GitHub repository contains only the code and non-restricted outputs. The SAM2 checkpoint file itself is not included because of file size. The checkpoint used was `sam2.1_hiera_large.pt`, with configuration file `sam2.1/sam2.1_hiera_l.yaml`.

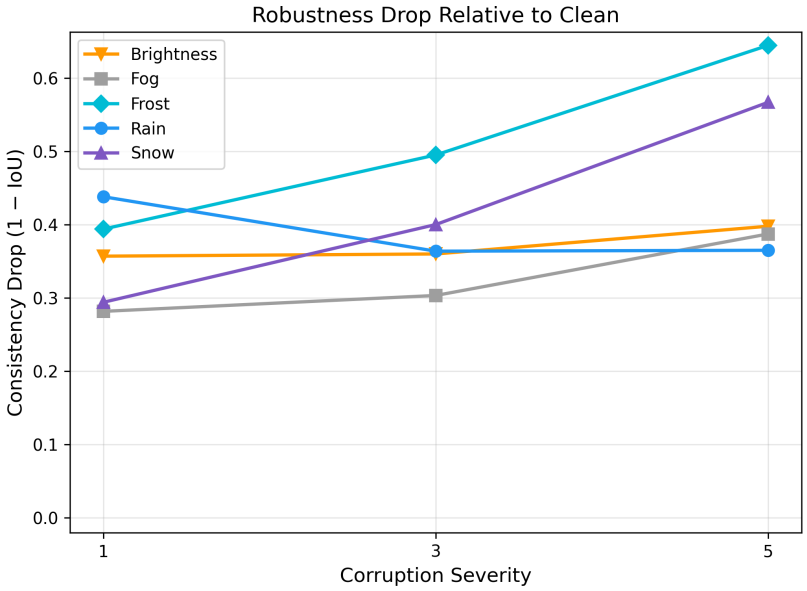



Figure B.1: Consistency drop relative to clean output across corruption severity levels. Higher values indicate greater divergence between clean and corrupted SAM2 foreground predictions.

## B.3 Final Implementation Scripts

Table B.1: Final implementation scripts and their roles.

| Script | Purpose |
|---|---|
| `run_sam2.py` | Runs SAM2 automatic mask generation on clean and corrupted images, saving raw `.npz` masks, qualitative overlays, logs, and run metadata. |
| `sam_eval_utils.py` | Provides shared utilities for loading SAM2 masks, building dense foreground maps, resizing masks, and computing overlap metrics. |
| `compute_consistency.py` | Computes the primary clean-vs-corrupted foreground consistency metrics. |
| `analyze_consistency.py` | Generates summary tables and quantitative plots for the consistency results. |
| `compute_iou.py` | Performs the supplementary GT-based best-overlap sanity check. |
| `make_qual_grids.py` | Creates qualitative comparison grids for report figures. |
| `make_rep_qual_grid.py` | Creates representative input-corruption and SAM2 overlay grids for selected scenes. |
| `test_sam_eval_utils.py` | Contains sanity tests for IoU, Dice, and nearest-neighbour mask resizing. |

The scripts `apply_weather.py` and `analyze_results.py` are retained only as legacy exploratory scripts. They were not used to produce the final reported Nano Banana Pro corruption set or the final consistency plots.

## B.4 Implementation Environment

Table B.2: Implementation environment used for the final SAM2 experiments.

| Item | Configuration |
|---|---|
| Operating system | Windows 11 |
| Machine | Lenovo Legion laptop |
| CPU | Intel Core i7-14700HX |
| GPU | NVIDIA GeForce RTX 5070 Laptop GPU |
| Memory | 16 GB RAM |
| Python environment | Conda environment named `sam2` |
| SAM2 checkpoint | `sam2.1_hiera_large.pt` |
| SAM2 configuration | `sam2.1/sam2.1_hiera_l.yaml` |
| Main inference profile | `balanced` |
| Final experiment scale | 5 images, 5 visual corruption conditions, 3 severity levels, giving 75 clean-corrupted pairs |

## B.5 External Resources Used

The project used the following main external resources:

- **Cityscapes**: source dataset and ground-truth labels used for the selected urban driving scenes [3].
- **SAM2**: pretrained foundation segmentation model used for automatic mask generation, with the official SAM2 repository used as the source for code, configuration, and checkpoint guidance [15, 13].
- **Nano Banana Pro / Gemini image generation**: image-to-image editing workflow used to create the final reported adverse-weather and brightness-related corrupted images [6, 5].
- **Google Colab**: used during development for testing and checking some plotting workflows [7].
- **draw.io / diagrams.net**: used to create the methodology workflow diagram included in the report [4].
- **Python libraries**: NumPy, OpenCV, pandas, matplotlib, tqdm, PyTorch, and supporting SAM2 dependencies. PyTorch was the main deep-learning framework used by SAM2 [14].
- **Albumentations**: used only for the early exploratory corruption-generation prototype, not for the final reported corruption set [1].

These resources are cited in the bibliography where they are discussed in the main report.