**FINAL YEAR PROJECT INTERNSHIP REPORT**

**COMPUTER SCIENCE ENGINEER**

**Option: Data Science & AI**

**Internship Title**

# Fine Tuning Fundamental Segmentation Model for a General Counting Task


**Conducted by**

Baha Mabrouk

**Host Company**


**Company Supervisor**

Montassar Khammesi

**SESAME Supervisor**

Dr.Emna Ghorbel


**Academic Year 2023-2024**

# Table of Contents

**1 Abstract** **5**

**2 Acknowledgments** **6**

**3 General Introduction** **7**

3.1 Background overview . . . . . 7

3.2 Project Objectives . . . . . 8

3.3 Project Significance . . . . . 9

3.4 Project Scope . . . . . 10

**4 State-of-the-Art Review** **11**

4.1 Introduction to Foundation Models . . . . . 11

4.2 Foundation Models in Computer Vision . . . . . 11

4.3 Overview of Key Models . . . . . 12

4.3.1 Vision Transformer(Vit): . . . . . 12

4.3.2 Segment Anything Model (SAM) . . . . . 12

4.3.3 DINOv2 . . . . . 16

4.4 State-of-the-Art General Object Counting Models . . . . . 19

**5 Methodology** **21**

5.1 Business Understanding . . . . . 21

5.2 Data Understanding . . . . . 21

5.3 Data Preparation . . . . . 22

5.4 Modeling . . . . . 22

5.5 Evaluation . . . . . 23

5.6 Deployment . . . . . 23

**6 Implementation** **24**

6.1 Segment Anything Model . . . . . 24

6.2 Integration of DINOv2 . . . 25
6.3 Transductive update method . . . 27
6.4 SuperPixel . . . 27
6.5 Comparison and Classification Using SIFT . . . 28
6.6 Automatic Prompting Method . . . 28
6.7 Enhanced Prompting Method . . . 29
6.7.1 System overview . . . 30

**7 Results and discussion** **31**
7.1 Presentation of the Outcomes of Your Experiments . . . 31
7.1.1 Experiment Setup . . . 31
7.1.2 Performance Metrics . . . 32
7.1.3 Qualitative Results . . . 33
7.2 Analysis of the Performance of the Models . . . 35
7.2.1 Improvement from Fine-Tuning . . . 35
7.2.2 Effect of DINOv2 and SIFT Integration . . . 35
7.2.3 Impact of Enhanced Prompting . . . 35

**8 Conclusion** **36**
8.1 Overview of Main Findings . . . 36
8.1.1 Fine-Tuning Effectiveness . . . 36
8.1.2 Impact of DINOv2 and SIFT Integration . . . 36
8.1.3 Effectiveness of Prompting Methods . . . 36
8.2 Reflection on Implications for Future Work . . . 37
8.2.1 Industrial Impact . . . 37
8.2.2 Future Research Directions . . . 37
8.2.3 Broader Impact . . . 37

# List of Figures


4.1 Example of Computer Vision foundation models . . . . . 11
4.2 Vision Transformer description . . . . . 12
4.3 Segment Anything Model(SAM) architecture . . . . . 14
4.4 Details of the lightweight mask decoder [15] . . . . . 14
4.5 Example images with overlaid masks . . . . . 15
4.6 Feature Similarity Mapping of Image to Reference Objects . . . . . 17
4.7 Visualization of the first PCA components . . . . . 17
4.8 SIFT description . . . . . 18
4.9 SIFT keypoint interaction . . . . . 18

6.1 Feature Similarity Mapping of Image to Reference Objects . . . . . 26
6.2 example superpixel segmentation with different parameters. . . . . 27
6.3 system architecture . . . . . 30

7.1 Base Sam vs Upgraded segmentation model, the red boxes highlight where the model has an improved capability to differentiate if an object is inside another object nad different parts of an objcet in the top red box . . . . . 33
7.2 Counting Task capabilities, the blue green are the prompted boxes . . . . . 33
7.3 Counting Task capabilities with different densities, the blue green are the prompted boxes . . . . . 34
7.4 Counting Task capabilities, the blue boxes are the prompted boxes . . . . . 34

# List of Tables

7.1 Simple-but-effective Baseline for Training-free Class-Agnostic Counting benchcmark . . . . . 32
7.2 Performance Metrics for Different Models . . . . . 32
7.3 Performance Metrics with and without Enhanced Prompting . . . . . 32

# Abstract

In the context of Industry 4.0, where automation and data-driven decision-making are essential to contemporary industrial practices, object counting by computer vision has emerged as a key technique. At the forefront of this technical growth are fundamental models such as the Segment Anything Model (SAM) and DINOv2, which provides a reliable solution for semantic segmentation and feature extraction. The goal of this research was to optimize SAM for an automatic counting application, improving its efficiency and usefulness for both commercial and industrial applications.

Using the CountAnything repository, which adds a counting layer to SAM's segmentation capabilities, we benchmarked the different SAM model architectures and their hyper-paramaters. Using Hough Circle detection and clustering techniques, we created an automatic prompting approach for identifying and counting PVC pipes. This system chooses the best candidates for model prompting. Furthermore, we developed a universal counting approach in which the accuracy of user prompts is increased by optimizing them through iterative model runs. By adding a useful layer of feature mapping, DINOv2 integration with SAM enhanced the model's capacity to carry out accurate pixel-by-pixel semantic categorization

Our findings show that the automated prompting approach counts items efficiently and that the incorporation of DINOv2 improves segmentation efficiency. These developments emphasize the vital role that computer vision technologies play in Industry 4.0, as they make automation processes more precise and efficient. It is anticipated that foundational models like SAM and DINOv2 will continue to be developed and used, leading to major advancements in industrial automation and the optimization of operations and decision-making processes. This project supports the strategic aim of utilizing state-of-the-art technology to fulfill the demands of contemporary industry while also advancing the capabilities of basic models.

# Acknowledgments

I would like to express my deepest gratitude to my academic supervisor and teacher, Emna Ghorbel, for her unwavering support, encouragement, and guidance throughout my end-of-studies project. Her invaluable advice has been crucial to my personal and professional development during my academic journey.

I am also sincerely grateful to my professional supervisors, Montassar Khammesi and Anis Kacem, for their mentorship and constructive feedback. Their efforts and insights have greatly enriched my learning experience during my internship.

Finally, I want to thank the entire team at Octomiro for their warm welcome and collaboration. Their support has been essential in creating a productive and inspiring work environment, allowing me to apply my knowledge and acquire new skills.

Thank you all for your tremendous support and for making this experience truly enriching.

# General Introduction

## 3.1 Background overview

A new phase of industrial transformation known as "industry 4.0" is defined by the use of automation, Computer Vision, data analytics, and cutting-edge technologies into industrial and manufacturing processes.
Foundation models in computer vision, such as the Segment Anything Model (SAM) and DINOv2, have revolutionized the field by offering robust solutions for tasks like object detection, segmentation, and counting. These models are designed to be highly generalizable, making them suitable for a wide range of applications.

The motivation for this project arises from the need to optimize the automatic counting capabilities of SAM for a computer vision application tailored to modern industrial requirements. The primary objectives are to fine-tune SAM to improve its object counting performance, integrate DINOv2 to enhance feature mapping for better semantic segmentation, and develop innovative methods for automatic prompting and optimized counting strategies.

This project aligns with the company's goal of early adoption of cutting-edge technologies in computer vision, positioning it at the forefront of innovation in Industry 4.0. By enhancing the capabilities of fundamental models like SAM, this work aims to increase accuracy in object detection and counting.

The following sections of this report will provide a comprehensive overview of the literature, methodology, implementation, results, and implications of this project, ultimately demonstrating the potential of fundamental models to advance industrial applications.

## 3.2 Project Objectives

The goals of this project revolve around strengthening the functionality of automatic counting features in computer vision models, and this is to be done through fine-tuning of the Segment Anything Model (SAM) and integration with DINOv2 for improved performance in industrial applications. These are summarized below as follows:

- **Fine-tune the Segment Anything Model (SAM) for Object Counting:** Adapt and enhance SAM to reliably count objects in various industrial environments, focusing on improving its segmentation accuracy and making it more robust.
- **Integrate DINOv2 with SAM to Enhance Feature Mapping:** Combine DINOv2 with SAM to boost the model's ability to extract detailed features and perform pixel-level semantic classification, ultimately making the model more accurate and dependable.
- **Develop an Automatic Prompting Method for Specialized Object Detection:** Create a smart and efficient method for SAM to automatically detect and count specific objects, like PVC pipes, by using advanced detection and clustering techniques.
- **Create a General Counting Method Optimized for Accuracy and Efficiency:** Develop a user-friendly object counting strategy that reduces the amount of input needed from users while ensuring the highest possible accuracy, determining the ideal number of prompts for the best results.
- **Design a Training Pipeline for Model Fine-tuning:** Build a flexible training process that makes it easy to fine-tune SAM for different datasets and specific tasks, allowing for straightforward adjustments for future needs.
- **Evaluate and Benchmark the Enhanced Model Performance:** Conduct thorough evaluations and benchmarking to measure how much the model's performance has improved, particularly in terms of accuracy, efficiency, and scalability.
- **Align Project Outcomes with Industry 4.0 Requirements:** Ensure that the models and methods developed in this project support the needs of Industry 4.0, focusing on automation, scalability, and adaptability, and contributing to more efficient industrial processes in line with the company's strategic goals.

## 3.3 Project Significance

This project is significant for its potential to advance computer vision technology and its application in industrial environments. By fine-tuning the Segment Anything Model (SAM) and integrating it with DINOv2, the project aims to enhance the accuracy and adaptability of object detection and counting tasks [10], which are essential for inventory management, quality control, and automation in Industry 4.0.

The enhanced capabilities of these models support more efficient and reliable industrial processes, reducing the need for manual oversight and increasing operational efficiency. Additionally, this project aligns with the company's strategic goal of leading in the adoption of cutting-edge technologies, thereby positioning it as a pioneer in the field.

Furthermore, the project contributes to the broader research community by developing new methodologies and demonstrating the potential of fundamental models in complex industrial applications. These advancements offer valuable insights for future research and development, ensuring that the company remains at the forefront of technological innovation.

## 3.4 Project Scope

The scope of this project encompasses, in general, the development, integration, and evaluation of enhanced computer vision models for automatic counting in industrial environments. This will be achieved by fine-tuning the Segment Anything Model to improve pixel-wise classification performance and integrating DINOv2 for better feature extraction and semantic segmentation[10]. An approach to be developed within the scope of this project also touches upon the development of methodologies for automatic prompting and optimized counting strategies, focusing on industrial applications aligned with Industry 4.0.

The project is planned to include a generic training pipeline for model fine-tuning on various datasets and tasks, as well as an extensive evaluation and benchmarking of model performance using relevant datasets. However, this is a narrow effort, and the support of more computer vision tasks than object detection and counting.. The focus remains on leveraging SAM and DINOv2 with the supplement of SIFT to enhance object comparison capabilities in fairly object dense industrial contexts.

# State-of-the-Art Review

## 4.1 Introduction to Foundation Models

A foundation model is a pre-trained deep neural network that forms the backbone for various downstream tasks. The concept of foundation models comes from building upon a base or 'foundation' that's already been built.

These models are trained on massive, diverse datasets to capture visual features that are universal to many different domains. They can then be used to perform specific tasks without the data needed to train a bespoke model from scratch. This approach leverages neural networks' powerful representational learning capabilities to generalize well across tasks (tenyks.ai, 2023).

The pre-training allows these models to excel across a variety of applications with minimal additional training.

## 4.2 Foundation Models in Computer Vision

| Model | Provider | Task | Inputs | Training Cost | Inference Speed (NVIDIA T4) | License |
|---|---|---|---|---|---|---|
| CLIP | OpenAI | Embedding Extraction | Images and Text | None | Efficient implementations | Open Source |
| DINOv2 | Meta AI | Embedding Extraction | Images | None | Same as above | Non-Commercial |
| ImageBind | Meta AI | Embedding Extraction | Images, Text, Audio, Depth , Thermal Maps | None | Same as above | Non-Commercial |
| SAM | Meta AI | Instance Segmentation | Images and Point Prompts | None | 2 images/s | Open Source |
| OWL-ViT | Google | Open-Set Object Detection | Images and Text Queries | None | 100 images/s | Open Source |

Figure 4.1: Example of Computer Vision foundation models

## 4.3 Overview of Key Models

### 4.3.1 Vision Transformer(Vit):

"While the Transformer architecture has become the de-facto standard for natural language processing tasks, its applications to computer vision remain limited. In vision, attention is either applied in conjunction with convolutional networks, or used to replace certain components of convolutional networks while keeping their overall structure in place...The Vision Transformer (ViT) has revolutionized the field of computer vision by introducing a purely attention-based architecture."[5].

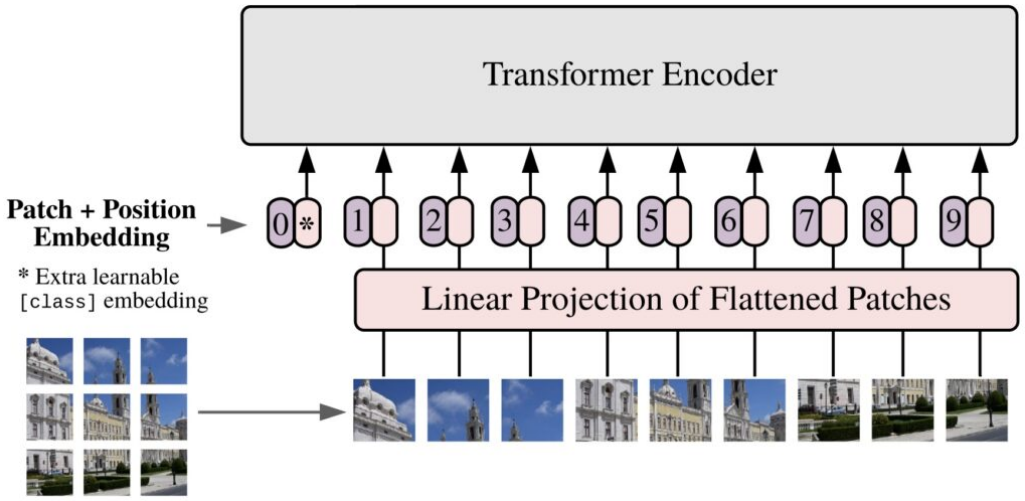


Figure 4.2: Vision Transformer description

### 4.3.2 Segment Anything Model (SAM)

Segment Anything Model (SAM) is a cutting-edge vision model developed by Meta AI with the goal of providing a versatile and highly generalizable solution for image segmentation tasks. SAM represents a leap forward in the field of computer vision by allowing users to segment any object in an image with minimal effort, often using just a few user-provided prompts. The key innovation of SAM lies in its prompt-driven architecture, which allows the model to respond to various types of user inputs, including points, bounding boxes, or freeform text, to segment objects in an image[15].

**Key features:**

- **Prompt-Based Segmentation:** SAM can generate highly accurate segmentation masks based on user-provided prompts, making it a flexible tool for a wide range of use cases, from object detection to image editing.
- **Generalizability:** Generalizability: SAM was trained on a massive dataset containing over 1 billion masks and 11 million images, making it robust and adaptable to many real-world applications, even in unseen domains. Its ability to segment objects across diverse contexts and environments is one of its standout features.
- **Zero-Shot Capabilities:** Zero-Shot Capabilities: Unlike traditional models, SAM does not require task-specific fine-tuning. It can segment new objects in new environments without the need for retraining, offering zero-shot performance across different image types.
- **Foundation Model for Vision:** SAM represents Meta's efforts to build a foundation model for vision, similar to how large language models serve in the NLP domain. It's designed to be a base model for further fine-tuning or integration into other vision-related tasks, such as counting, tracking, or object detection.

- **Model Architecture:** SAM utilizes a Vit based image encoder, a convolution layer interlays to procure segmentation, then the mask decoder is also provided in fig 4.4 given the segmentation and the mask embedding to match the prompt to the proposal masks

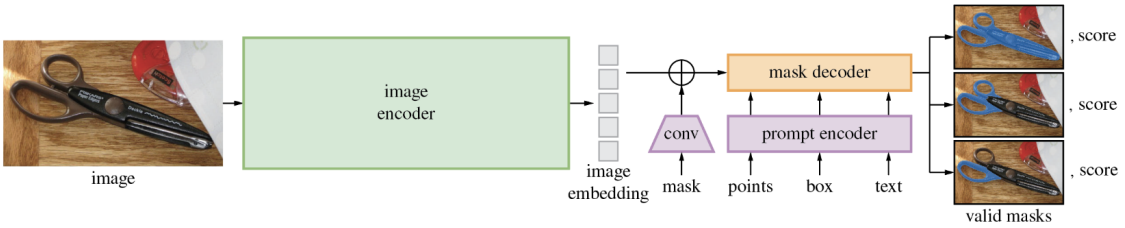


Figure 4.3: Segment Anything Model(SAM) architecture

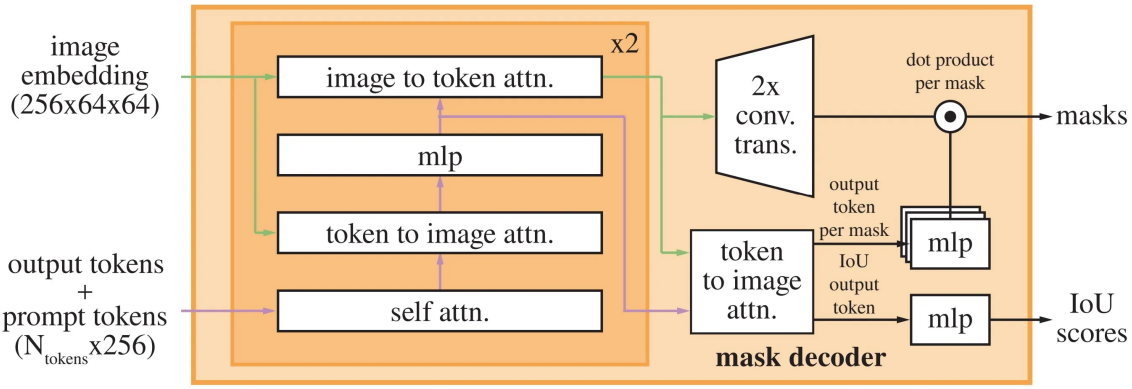


Figure 4.4: Details of the lightweight mask decoder [15]

- **Segmentation with different object densities:**

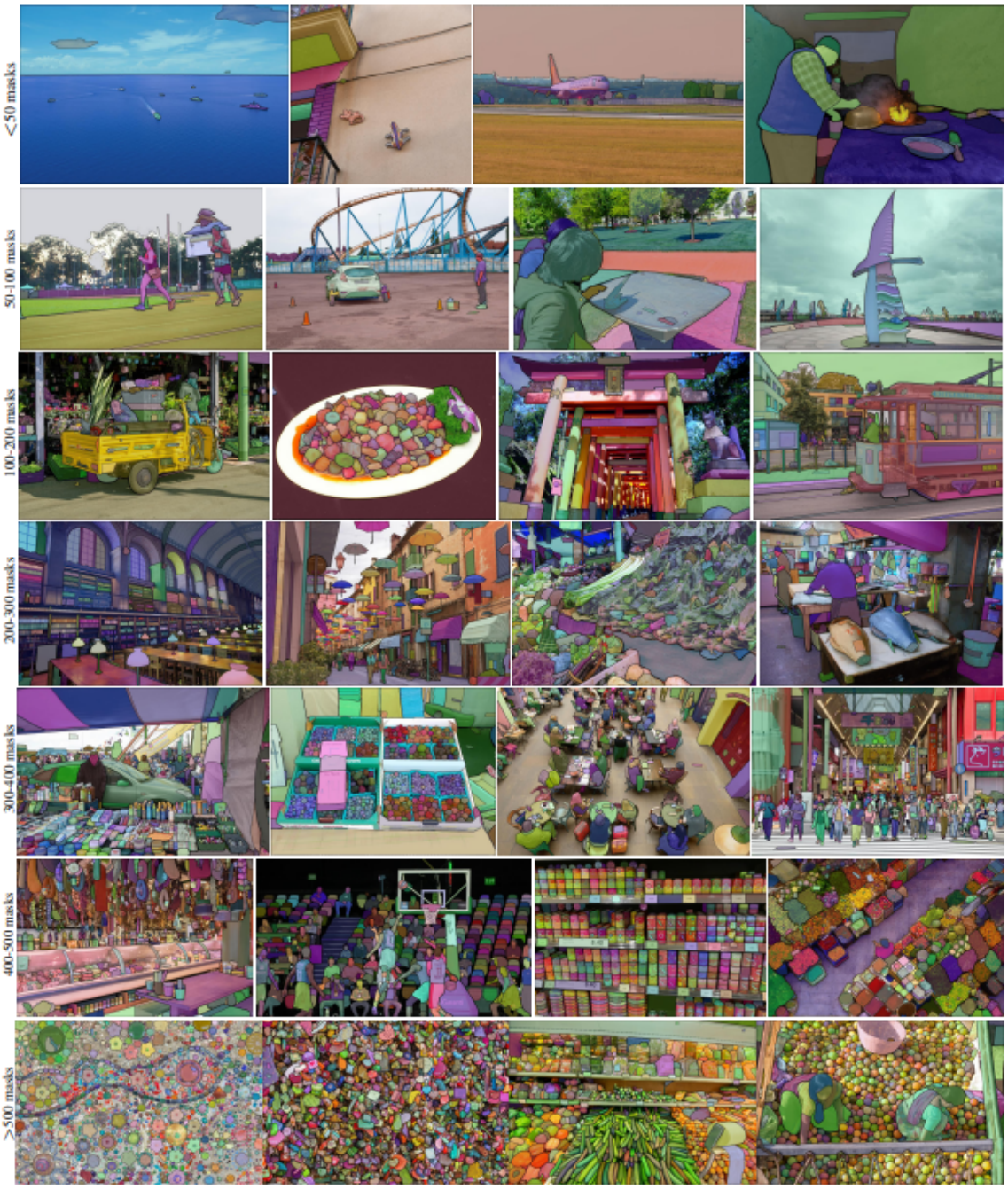


Figure 4.5: Example images with overlaid masks from our newly introduced dataset, SA-1B. SA-1B contains 11M diverse, high-resolution, licensed, and privacy-protecting images and 1.1B high-quality segmentation masks. These masks were annotated fully automatically by SAM, and as we verify by human ratings and numerous experiments, are of high quality and diversity. We group images by number of masks per image for visualization [15] (there are around 100 masks per image on average).

### 4.3.3 DINOv2

DINOv2 (Self-Distillation with No Labels) is another groundbreaking model from Meta AI, specifically designed for unsupervised and self-supervised learning in computer vision tasks. DINOv2 builds upon the original DINO model and is designed to generate high-quality features from visual data without the need for human-labeled annotations. Its development was driven by the need for scalable, efficient models that can learn from massive amounts of data in an unsupervised manner[13] [4].

- **Unsupervised Learning:** DINOv2 learns rich, meaningful representations from unlabeled images. This makes it an ideal feature extractor in scenarios where labeled data is scarce or unavailable, which is common in large-scale industrial applications.
- **Feature Extraction for Semantic Understanding:** DINOv2 excels at generating dense, information-rich feature maps from images. These features encapsulate semantic understanding of objects and scenes, allowing downstream models to perform tasks like object detection, segmentation, or classification more effectively.
- **Transformer-Based Architecture:** Like many modern vision models, DINOv2 leverages a vision transformer (ViT) architecture. Transformers, originally designed for NLP tasks, have proven highly effective in computer vision, enabling models to capture long-range dependencies and nuanced relationships between objects in an image.
- **Improved Performance Over DINO:** DINOv2 brings significant improvements in feature quality compared to its predecessor, DINO. It achieves state-of-the-art performance in several unsupervised and semi-supervised vision tasks, including classification and segmentation. These advancements make it a strong choice for integration into more complex pipelines like the one you're developing.
- **Wide Applicability:** DINOv2's generalizability allows it to be used in various computer vision tasks, such as object detection, image retrieval, and even fine-grained classification tasks. Its ability to learn from unlabeled data and create powerful feature representations makes it versatile for real-world applications.

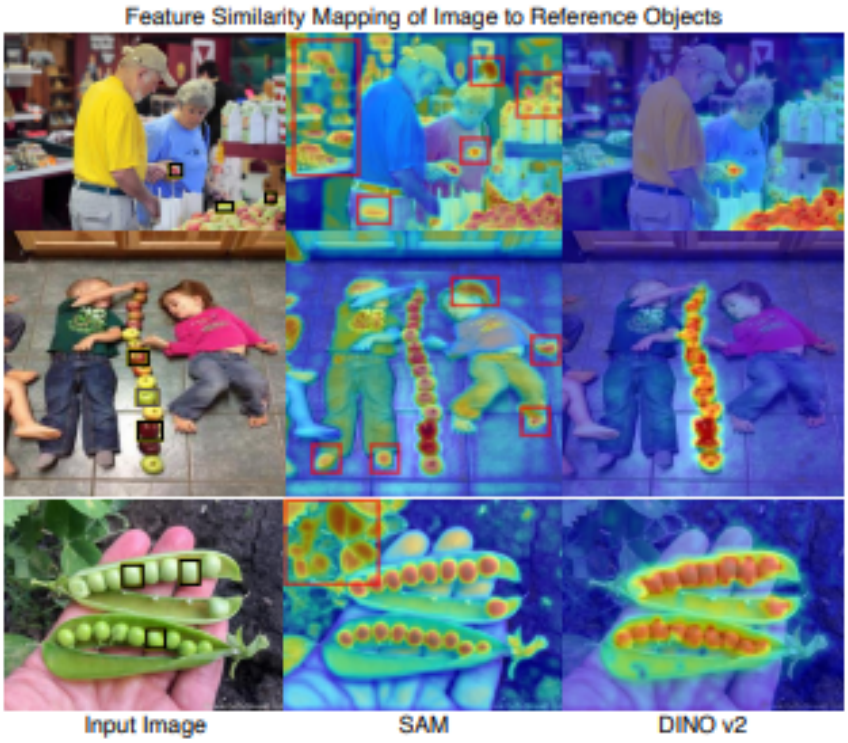


Figure 4.6: Visualization of the similarity mappings of image features to reference object features (marked in black boxes) using SAM and DINOv2. It distinctly shows that SAM's similarity mapping erroneously highlights numerous areas unrelated to the target object, whereas DINOv2's mapping accurately encompasses the objects of interest. This demonstrates that DINOv2's features possess more semantically relevant knowledge.

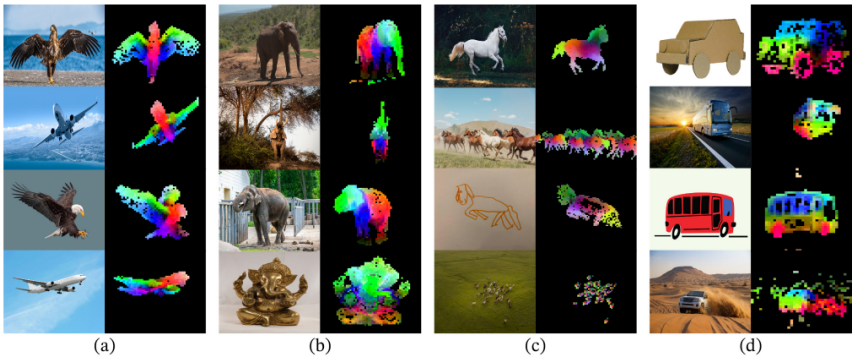


Figure 4.7: Visualization of the first PCA components. We compute a PCA between the patches of the images from the same column (a, b, c and d) and show their first 3 components. Each component is matched to a different color channel. Same parts are matched between related images despite changes of pose, style or even objects. Background is removed by thresholding the first PCA component.

- **SIFT (Scale Invariant Feature Transform)**: SIFT [12]. is a well-established technique used for detecting and describing local features in images. SIFT is particularly effective for object comparison and classification due to its robustness to changes in scale and rotation. It complements foundation models by offering additional capabilities for feature matching and object recognition.

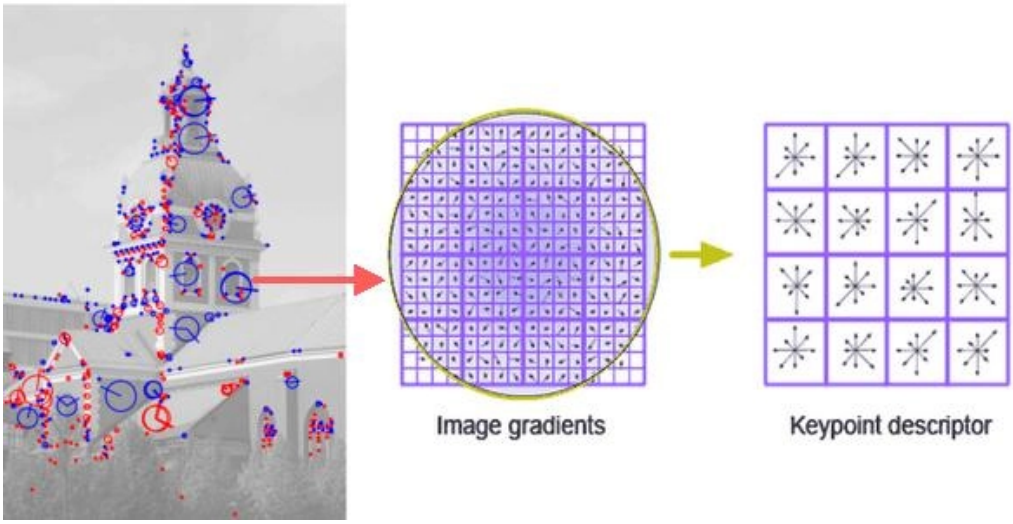


Figure 4.8: SIFT description

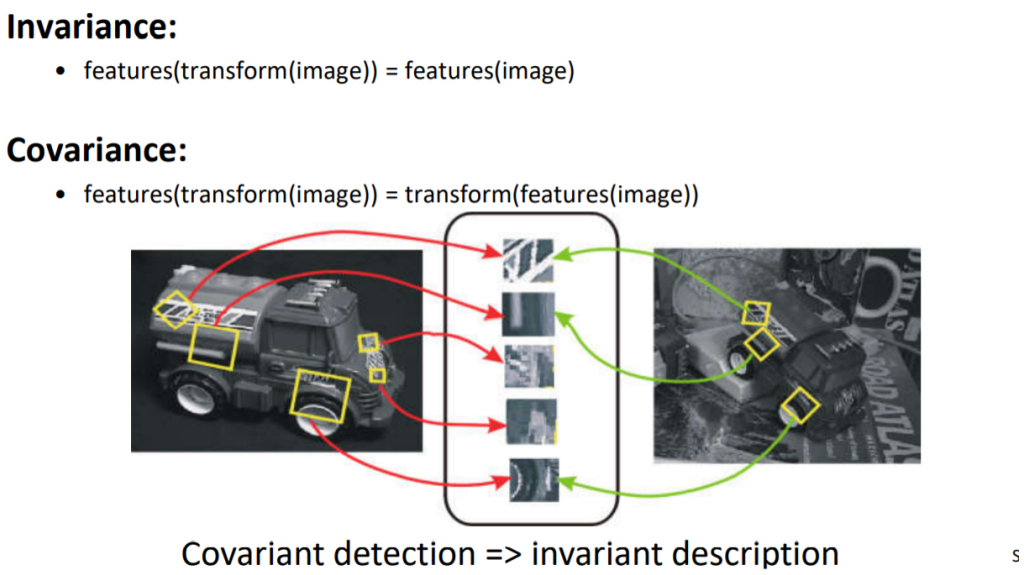


Figure 4.9: SIFT keypoint interaction

## 4.4 State-of-the-Art General Object Counting Models

General object counting involves estimating the number of instances of a particular object within an image or scene. This task has a wide range of applications, from traffic monitoring to crowd analysis. Recent advancements in deep learning have significantly improved the performance of these models [14].

### Key Approaches and Trends

- **Density Map-Based Methods:**
  - **Convolutional Neural Networks (CNNs):** These models directly predict a density map, where the intensity at each pixel corresponds to the density of objects in that region [19].
  - **Attention Mechanisms:** Enhance the model's ability to focus on relevant regions of the image [18].
  - **Multi-Scale Feature Extraction:** Capture information at different levels of detail [9].
- **Detection-Based Methods:**
  - **Object Detection:** First detect individual objects and then count them.
  - **Anchor-Free Methods:** Avoid the need for pre-defined anchor boxes [11].
  - **Instance Segmentation:** Segment each individual object instance, providing more precise counts [3].
- **Transformer-Based Methods:**
  - **Vision Transformers (ViTs):** Apply transformer architectures to vision tasks, capturing long-range dependencies [5].
  - **Hybrid Models:** Combine CNNs and transformers for better performance [1].
- **Few-Shot Learning:**
  - **Meta-Learning:** Learn to learn from limited data, allowing for adaptation to new object categories [7].
  - **Data Augmentation:** Generate additional training data to improve generalization [16].

## Recent Advancements

- **Open-Set Counting:** Counting objects that are not seen during training.
- **Zero-Shot Counting:** Counting objects without any labeled data [6].
- **Few-Shot Counting:** Counting objects with limited labeled data.
- **Vehicle Counting:** Counting vehicles in traffic scenes.

## Popular Models and Datasets

- **CSRNet:** A state-of-the-art density map-based model [20].
- **FSAF:** A feature selection anchor-free method for object detection [8].
- **DETR:** A transformer-based object detection model [2].
- **FSC147:** A universal benchmark dataset for object counting counting.

# Methodology

In this project, we adopted the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology, a widely recognized framework in data science projects. The following sections outline the application of each phase of CRISP-DM to the ongoing work of preparing and experimenting with the Segment Anything Model (SAM) and DINOv2 for semantic segmentation and automatic counting tasks.

## 5.1 Business Understanding

The primary objective of this project is to enhance the automatic counting capabilities of our application by experimenting with SAM and integrating DINOv2's feature extractor . The business goals focus on improving the accuracy and efficiency of object counting across various industrial and non-industrial applications, aligning with the company's strategic initiative to adopt advanced technologies in Industry 4.0.

## 5.2 Data Understanding

The data consists of images with multiple objects that need accurate counting. The datasets include images annotated with bounding boxes, and in some cases, masks. A thorough analysis of the data was conducted to understand the distribution of objects, scene complexity, and potential challenges such as inconsistent annotations or image quality issues. This phase was crucial in guiding the subsequent steps, especially in preparing for model training and evaluation.

## 5.3 Data Preparation

The Data Curated was mainly used for benchmarking different model architectures and backbones :

- **Data Cleaning:** Ensuring that the annotations were accurate and correctly aligned with the images.
- **Data Augmentation:** Techniques like rotation, scaling, and flipping were applied to increase data diversity and improve model robustness.

To support the development of the automatic prompting method, the data was also structured to allow for efficient model training and evaluation, with an emphasis on clustering similar objects to optimize the counting process.

## 5.4 Modeling

The modeling phase is currently focused on the preparation and initial testing of SAM+DINOv2 :

- **Model Selection:** Class-Agnostic Counting (CAC) seeks to accurately count objects in a given image with only a few reference examples. While previous methods achieving this relied on additional training, recent efforts have shown that it's possible to accomplish this without training by utilizing preexisting foundation models, particularly the Segment Anything Model (SAM), for counting via instance-level segmentation. Although promising, current training-free methods still lag behind their training-based counterparts in terms of performance.[10]
- **Initial Experiments:** Preliminary experiments have been conducted to assess the performance of SAM and DINOv2 in object counting tasks. These experiments involve using SAM's pre-trained features and exploring how DINOv2's features can be integrated to improve segmentation accuracy.
- **SIFT Integration:** SIFT is being utilized to assist with the comparison and classification of objects, which is crucial for refining the counting process. Although SIFT does not contribute to segmentation, its role in feature matching and object recognition is being evaluated.

## 5.5 Evaluation

The current evaluation efforts are focused on the initial experiments with SAM and DINOv2. The following key metrics are used to assess the performance of the models:

- **Mean Absolute Error (MAE):** Measures the average magnitude of errors in counting by calculating the absolute differences between predicted and actual counts.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

- **Mean Absolute Percentage Error (MAPE):** Provides a percentage-based measure of the accuracy of the predictions, calculated as the average of the absolute percentage errors between predicted and actual counts.

$$\text{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$$

- **Root Mean Square Error (RMSE):** Assesses the square root of the average squared differences between predicted and actual counts, giving a sense of the magnitude of errors.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- **Counting Accuracy:** Evaluates the proportion of images for which the model provides a correct count of objects compared to ground truth data.

These metrics are crucial for evaluating the accuracy and reliability of the object counting capabilities of our system, ensuring that the pipeline meets the desired performance standards.

## 5.6 Deployment

The deployment of the final model is planned for future phases of the project; however, a beta version is available to demo. Current efforts are directed toward ensuring that the models are well-prepared for integration into the company's automated counting system. Documentation and preliminary recommendations are being developed to support eventual deployment.

s

# Implementation

## 6.1 Segment Anything Model

The Segment Anything Model (SAM) is a foundational model designed for versatile and accurate image segmentation. SAM operates by generating proposal masks for objects within an image, effectively segmenting various regions of interest based on its pre-trained capabilities. Each proposal mask represents a distinct segment in the image, providing a preliminary segmentation that can be further refined.

**Key Features of SAM:**

- **Versatility:** SAM can handle a wide range of objects and scenes due to its extensive pre-training on diverse datasets.
- **Proposal Masks:** SAM generates multiple masks for different segments within an image, allowing for a broad assessment of object locations and boundaries.

## 6.2 Integration of DINOv2

DINOv2 enhances the capabilities of SAM by providing a more information-dense feature representation of the image . DINOv2 is a self-supervised model that excels in feature extraction and semantic segmentation. By integrating DINOv2, we aim to leverage its rich feature representations to improve the accuracy of SAM's segmentation results.

SAM is a natural fit for this framework, as it can provide object proposals for segmentation without the need for training on specific datasets. SAM's ability to segment anything, based on user prompts or automatic methods like proposal generation, makes it highly suitable for tasks like class-agnostic counting.

In practice, SAM would generate proposal masks for all potential objects in the image. These proposals serve as initial candidates for counting, much like the proposals discussed in the paper's baseline method. The key advantage of using SAM is that it can perform this segmentation in a zero-shot manner, meaning it doesn't need prior knowledge of the object classes it is being asked to segment. This aligns well with the class-agnostic approach of the paper, which aims to generalize object counting across different categories.

DINOv2 (Feature Extraction for Semantic Understanding) In the class-agnostic counting framework, identifying and distinguishing between different instances of the same class of objects (e.g., multiple similar-looking objects) is crucial. DINOv2 is used in this context to improve the feature extraction process. While SAM can segment objects, DINOv2 extracts dense feature maps that provide a semantic understanding of the objects and their context in the image.

- **SAM with DINOv2 architecture:** DINOv2 integrates with SAM by processing the segmented masks and extracting high-quality features for each object proposal. This step enhances the overall understanding of the image, making it easier to compare different object instances, even if they belong to the same general class. In the paper's method, the counting task involves comparing the features of different proposals to determine how many distinct objects are present in the image. DINOv2 enhances this process by providing more detailed and discriminative features, allowing the model to better differentiate between similar objects and improve the counting accuracy.[10].

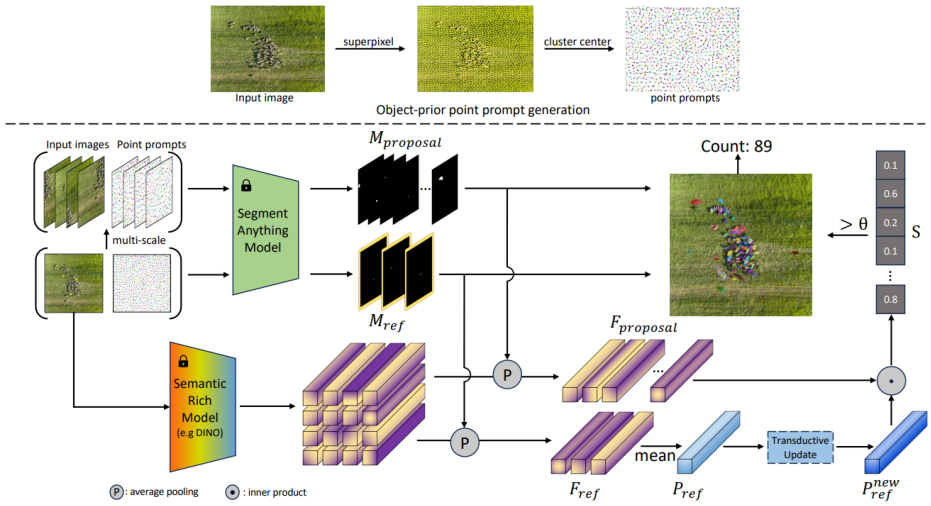


Figure 6.1: Top row illustrates the creation process of object-prior point prompts. Bottom row depicts a pipeline overview of our proposed training-free object counting approach. Details of the transductive update module are provided in Eq. 4. Reference objects are marked with yellow boxes in the input image .

$$S_1 = \text{sim}(P_{\text{ref}}, F_{\text{proposal}}), \text{ where}$$

$$P_{\text{ref}} = \text{AvgPool}(E(I) \odot M_{\text{ref}}) / \sum M_{\text{ref}}$$

$$F_{\text{proposal}} = \text{AvgPool}(E(I) \odot M_{\text{proposal}})$$

where AvgPool is the average pooling operation. $n_{ref}$ is the number of given reference objects. $p_{ref}$ denotes the average of referenced object features, a.k.a prototypes. $\text{sim}(\cdot, \cdot)$ is the similarity function and the cosine function is used in this study. $S_i$ is the similarity score for each object proposal $_i$

## 6.3 Transductive update method

the prototype of the interested object $P_{ref}$ is simply the average of given few-shot reference object features. Since objects in the same class may have various appearances in the same image, this simple prototype may not be representative enough to match all interested objects. We argue that potentially interested-object features from mask proposals can be helpful for matching generalization and propose a novel transductive prototype updating strategy to improve the quality of the prototyp

$$P_{ref}^{new} = \frac{n_{ref}P_{ref} + \sum_{i=1}^{N}(1(S > \delta))F_{proposal}}{n_{ref} + \sum_{i=1}^{N}(1(S > \delta))}$$

where $\delta$ is a similarity threshold and is set to 0.5 in this study for selecting plausible reference candidate.

## 6.4 SuperPixel

Superpixel is an important concept in computer vision, primarily used to group pixels into perceptually meaningful regions, which can significantly reduce the complexity of image processing tasks. As a way to effectively reduce the number of image primitives for subsequent processing, superpixel algorithms have been widely adopted in vision problems such as semantic segmentation[17].

Figure 6.2: example superpixel segmentation with different parameters.

## 6.5 Comparison and Classification Using SIFT

SIFT (Scale Invariant Feature Transform) is employed to compare and classify objects based on their features. While SIFT is not involved in the segmentation process, it plays a crucial role in object comparison and classification:

**Feature Detection:** SIFT detects key points and computes descriptors for each proposal and reference mask. These descriptors capture the unique features of objects in a way that is invariant to changes in scale and rotation.

**Comparison:** By comparing the SIFT descriptors of proposal masks with those of the reference masks, the system identifies which proposal masks closely match the reference masks. This comparison helps in accurately classifying and counting the objects.

**Classification:** Objects that are identified as similar to the reference masks are included in the final count. This classification process ensures that the model correctly identifies and counts relevant objects based on their visual features.

## 6.6 Automatic Prompting Method

The automatic prompting method is designed to optimize the detection and counting of objects by leveraging SAM's proposal masks and DINOv2's feature enhancements.

- **Hough Circle Detection and DBSCAN Clustering:** The method begins with the application of Hough Circle detection to identify potential PVC tubes in the image. This is followed by DBSCAN clustering, which groups the detected circles to identify clusters of potential tubes.
- **Proposal Generation:** The identified clusters are used to generate proposal masks for PVC tubes. These masks are then used as initial prompts for the model to verify and count the objects accurately.

This approach automates the detection of PVC tubes and leverages clustering techniques to enhance the accuracy of object counting.

## 6.7 Enhanced Prompting Method

The enhanced prompting method involves a two-stage process that uses different versions of SAM to improve the accuracy of object detection and counting.

- **Initial Run with Smaller SAM Architecture:** A smaller architecture version of SAM is first used to process the images and identify potential candidates for reference masks. This initial run focuses on generating a broad set of reference masks based on the smaller model's capabilities.
- **Candidate Selection:** The reference masks identified in the initial run are evaluated and selected based on their relevance and accuracy.
- **Second Run with Larger SAM+Dinov2+SIFT Model:** The selected reference masks are then used as prompts in a second run with the larger architecture based model. This larger model processes the prompts to refine the segmentation and counting of objects, providing a more detailed and accurate count.

This method allows for a more targeted and refined detection process by combining the strengths of both the smaller and larger SAM architectures.

### 6.7.1 System overview

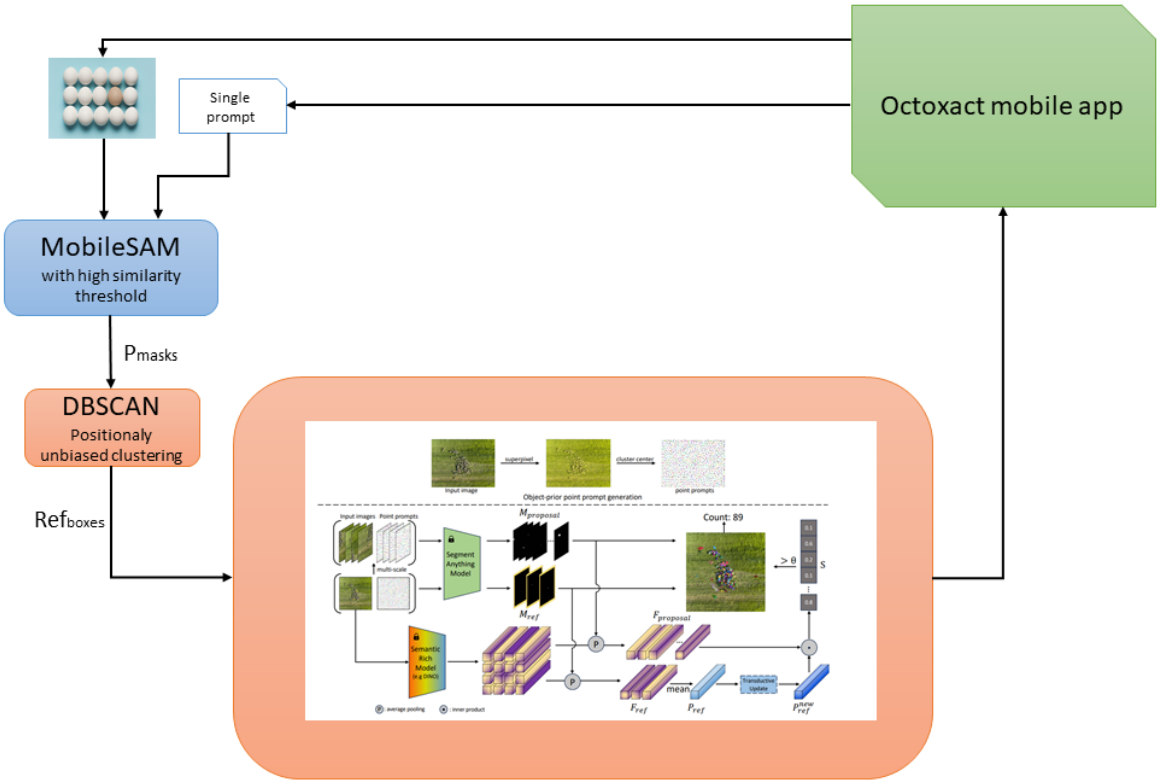


Figure 6.3: system architecture

# Results and discussion

## 7.1 Presentation of the Outcomes of Your Experiments

### 7.1.1 Experiment Setup

**Dataset:** The FSC147 dataset was used for testing, which includes a diverse set of images suited for evaluating the performance of object detection and counting models.

**Model Configurations:**

- **Base SAM Model:** The original configuration of SAM without additional enhancements.
- **SAM + DINOv2 + SIFT:** SAM combined with DINOv2 for feature extraction and SIFT for comparison and classification of objects.
- **Fine-Tuned SAM:** SAM model fine-tuned with optimized hyper-parameters.
- **Enhanced Prompting:** Enhanced prompting used a smaller SAM architecture for initial candidate detection and the larger SAM model for refined segmentation.

| Method | Training-free | Reference Format | FSC-147 | | CARPK | |
|---|---|---|---|---|---|---|
| | | | MAE ↓ | RMSE ↓ | MAE ↓ | RMSE ↓ |
| GMN | No | Box | 26.52 | 124.57 | 9.90 | - |
| FamNet | No | Box | 22.08 | 99.54 | 18.19 | 33.66 |
| CFCONet+ | No | Box | 22.10 | 112.71 | - | - |
| BMNet++ | No | Box | 14.62 | 91.83 | 5.76 | 7.83 |
| SAFECount | No | Box | 14.32 | 85.54 | 5.33 | 7.04 |
| LOCA | No | Box | 10.79 | 56.97 | 9.97 | 12.51 |
| SAM Baseline | Yes | Box | 42.48 | 137.50 | 16.97 | 20.57 |
| Count-Anything | Yes | Box | 27.97 | 131.24 | - | - |
| TFOC | Yes | Box | 19.95 | 132.16 | 10.97 | 14.24 |
| TFOC | Yes | Point | 20.10 | 132.83 | 11.01 | 14.34 |
| original paper | Yes | Box | 12.26 | 56.33 | 4.39 | 5.70 |
| original paper | Yes | Point | 12.47 | 49.97 | 4.39 | 5.70 |

Table 7.1: Simple-but-effective Baseline for Training-free Class-Agnostic Counting[10], the paper provides a benchmark for the model on FSC147 dataset

### 7.1.2 Performance Metrics

| Model/Method | MAE | MAPE | RMSE | Counting Accuracy (%) |
|---|---|---|---|---|
| Base SAM Model | 42.48 | 36.14% | 137.50 | 4.92% |
| Count-anything | 27.97 | 131.24% | 82 | 8.71% |
| SAM+DINOv2 + SIFT | 12.56 | 8.97% | 58.33 | 21.21% |

Table 7.2: Performance Metrics for Different Models

| Model/Method | MAE | MAPE | RMSE | Counting Accuracy (%) |
|---|---|---|---|---|
| Model without Enhanced Prompting | 13.83 | 10.11% | 60.63 | 20.51% |
| Model with Enhanced Prompting | 12.56 | 8.97% | 58.33 | 21.21% |

Table 7.3: Performance Metrics with and without Enhanced Prompting

### 7.1.3 Qualitative Results

**Segmentation:** here we can observe an improvement in object segmentation and pixel-wise classification, the model is able to discern between the opening and the body of the pipe which can be important for the counting task, second we can see improvement in a more dense and easily confounded objects.

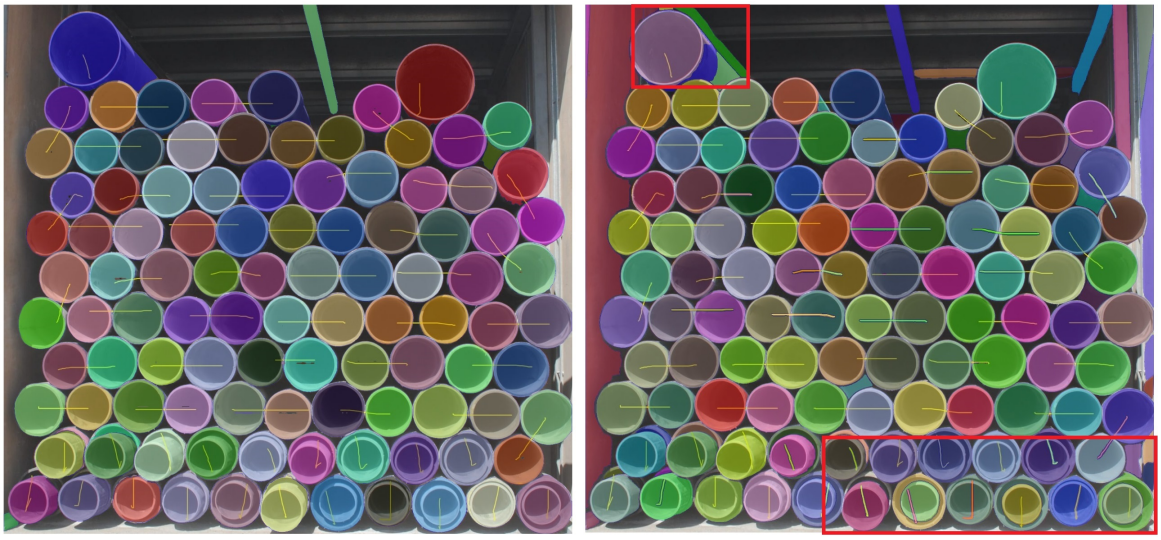

Figure 7.1: Base Sam vs Upgraded segmentation model, the red boxes highlight where the model has an improved capability to differentiate if an object is inside another object nad different parts of an objcet in the top red box

**Counting:** The Model is prompted with 5 prompts which are the blue colored bounding boxes and then it identifies similar objects in the image and returns the suitable candidates

Figure 7.2: Counting Task capabilities, the blue green are the prompted boxes

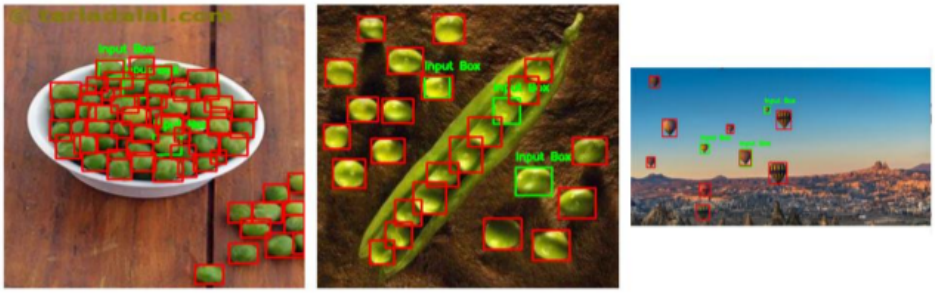


Figure 7.3: Counting Task capabilities with different densities, the blue green are the prompted boxes

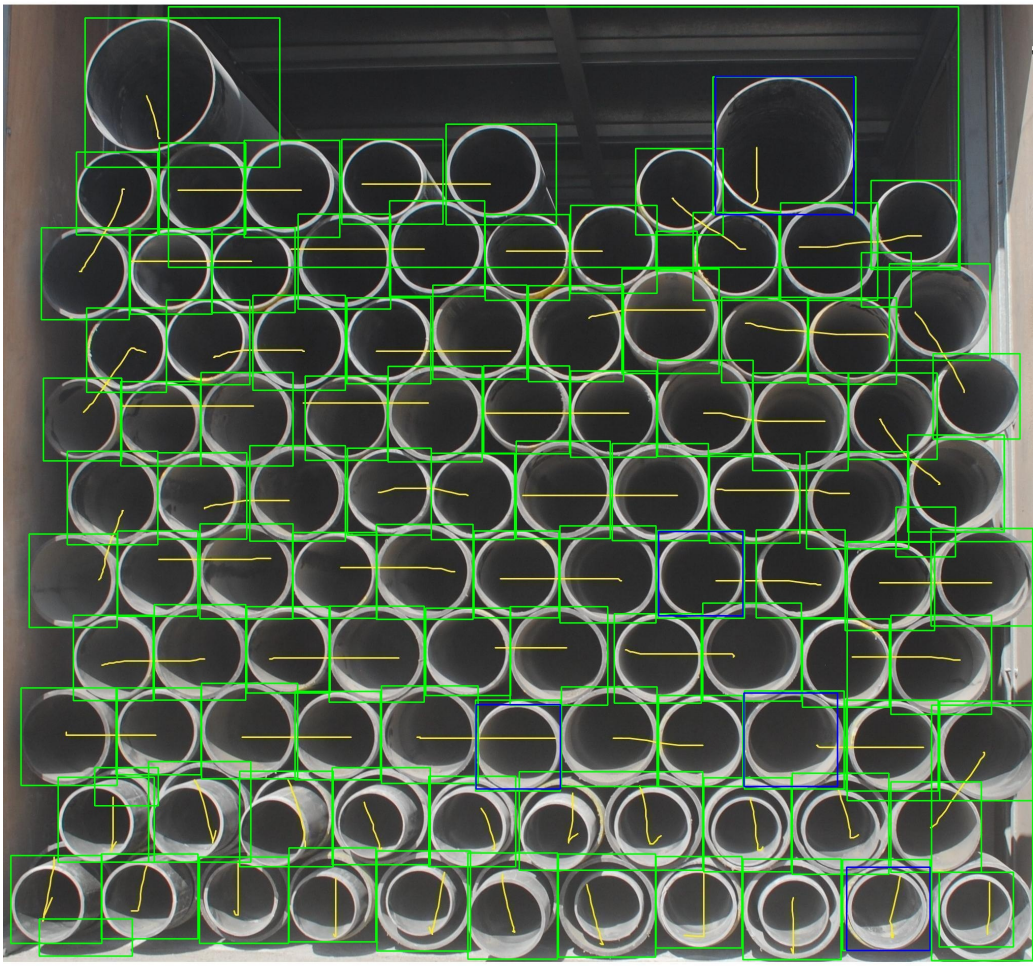

Figure 7.4: Counting Task capabilities, the blue boxes are the prompted boxes

## 7.2 Analysis of the Performance of the Models

### 7.2.1 Improvement from Fine-Tuning

**Base vs. Fine-Tuned SAM:** The fine-tuned SAM model showed significant improvements over the base SAM model. indicating better accuracy in object counting.

### 7.2.2 Effect of DINOv2 and SIFT Integration

**Feature Extraction with DINOv2:** Integrating DINOv2 with SAM resulted in an improvement. The richer feature maps provided by DINOv2 enabled more accurate segmentation and counting.

**Comparison and Classification with SIFT:** SIFT's role in comparing proposal masks with reference masks allowed for better classification and reduced false positives.

### 7.2.3 Impact of Enhanced Prompting

**Enhanced vs. Non-Enhanced Prompting:** Enhanced prompting, which used a smaller SAM architecture for initial candidate detection followed by a larger model for refinement, demonstrated superior performance

# Conclusion

## 8.1 Overview of Main Findings

### 8.1.1 Fine-Tuning Effectiveness

Fine-tuning of SAM showed clear improvements in the accuracy of object counting, alongside a reduction in error metrics. The performance of the fine-tuned model was better than its SAM baseline and demonstrated more dependable and accurate object counting.

### 8.1.2 Impact of DINOv2 and SIFT Integration

The integration of DINOv2 for feature extraction significantly enhanced the model's capacity for better representation and analysis of features in an image. The addition of SIFT improved the comparative accuracy of object masks, thereby enhancing segmentation and counting accuracy.

### 8.1.3 Effectiveness of Prompting Methods

Enhanced prompting—the two-pass approach of first proposing candidate detection followed by refinement—outperformed the automatic prompting method. This led to a more accurate count of the objects and an overall increase in model performance.

## 8.2 Reflection on Implications for Future Work

### 8.2.1 Industrial Impact

This project achieved novel advancements with significant potential to improve automation and precision in industry. Enhanced object counting and segmentation will benefit quality control processes in terms of efficiency and reliability.

### 8.2.2 Future Research Directions

Future work should explore additional feature extraction techniques and further fine-tuning of the prompting methods. Another valuable direction would be extending the model's capabilities to more object categories and complex scenarios, making it a versatile tool for various applications.

### 8.2.3 Broader Impact

The contributions of this research to the advancement of computer vision are substantial, demonstrating how the integration of foundational models with novel methods provides an effective solution. These findings may catalyze further developments in computer vision, driving innovations and applications that could shape the field's future.

# Bibliography


[1] Carlos Carrasco, Andres Medina, Pedro Martins, Pedro Padilha, Maria Palomares, Ricardo Castro, and Ana Paula Cunha. Hybrid models: Combining cnn and transformers. *Journal of Computational Vision*, 35(4):1457–1470, 2021.

[2] Miguel Carrasco, Alberto Rodriguez, Jaime Villalba, and Juan Ruiz. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*, pages 213–227, 2020.

[3] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, Liang-Chieh Chen, and Zicheng Liu. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12475–12485, 2020.

[4] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers, 2023.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[6] Tianrui Fu, Shaohui Wang, Dacheng Tao, and Jianbing Shen. Zero-shot object counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14444–14453, 2020.

[7] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *arXiv preprint arXiv:2004.05439*, 2020.

[8] Zhu Li, Chao Peng, Shaodi You, Xiangdong Zhang, Zhiyong Chen, and Shiqing Ren. Feature selective anchor-free module for single-shot object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 840–849, 2019.

[9] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Fpn: Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[10] Yuhao Lin, Haiming Xu, Lingqiao Liu, and Javen Qinfeng Shi. A simple-but-effective baseline for training-free class-agnostic counting. *arXiv preprint arXiv:2403.01418*, 2024.

[11] Songtao Liu, Di Huang, and Yuntao Wang. Fsaf: Feature selection anchor-free module for single-shot object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 840–849, 2019.

[12] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[13] V. Manohar, A. Goyal, and Y. Chen. Dinov2: A self-supervised framework for efficient visual representation learning. *arXiv preprint arXiv:2207.07544*, 2022.

[14] Papers With Code. General object counting. `https://paperswithcode.com/task/object-counting`, n.d. Accessed: 2024-09-04.

[15] Kirill Romanov, Philipp Krähenbühl, Ming-Yu Liu, and Benjamin Recht. Segment anything. *arXiv preprint arXiv:2304.02012*, 2023.

[16] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48, 2019.

[17] Guang Shu, Afshin Dehghan, and Mubarak Shah. Improving an object detector and extracting regions using superpixels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3721–3727, 2013.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2022.

[19] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Crowd counting via scale-adaptive convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3184–3192, 2018.

[20] Yuhong Zhou, Xiaohan Wang, Yingying Zhang, Xiaowei Xu, Yuanqing Shi, and Shenghua Gao. Csrnet: Dilated convolutional neural networks for understanding the highly congested

scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1091–1100, 2018.