STRUCTURED SPATIAL REASONING FOR ROBUST AND TRANSPARENT OBJECT COUNTING

by

Rishikesh Bhyri

May 1, 2026

A thesis submitted to the
faculty of the Graduate School of
the University at Buffalo, The State University of New York
in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computer Science and Engineering

Dedication

To the memory of my grandmother.

Acknowledgments

I would like to express my sincere gratitude to my advisor, Dr. Junsong Yuan, for his invaluable guidance during the course of my Master's program. His training helped me refine my research taste and mindset, enabling me to identify and solve important, impactful problems. I would also like to thank Dr. Nan Xi for his hands-on approach to mentoring me and for helping me articulate and present my research.

I extend my thanks to my committee members, Dr. Kaiyi Ji and Dr. Nan Xi, for their valuable feedback and insights, which have greatly improved this thesis. Additionally, I would like to thank Dr. Peter C.W. Kim, MD, for providing me with the opportunity to work on impactful research problems in the medical domain. He was instrumental in inculcating the mindset necessary for engineering safety-critical applications with a near-zero margin for error.

I am also deeply grateful to my collaborators, Dr. Brian R. Quaranto, MD, and Dr. Philip Seger, MD, for their continued support and collaboration.

Last but not least, I want to thank my family for being my backbone. Their unconditional support and motivation have helped me overcome countless obstacles. They have been my guiding light throughout my life, and without them, this achievement would not have been possible.

0.1 Disclaimer

This thesis is a compilation of research conducted during my Master of Science studies at the University at Buffalo. Portions of the research presented in this document are based on collaborative works that have been published at WACV [1] or submitted to peer-reviewed venues, for which I served as the primary first author.

Because these chapters are adapted from collaborative multi-author manuscripts, the plural pronoun "we" has been retained throughout the text to acknowledge the invaluable guidance of my advisor and the efforts of my co-authors. However, the core methodologies, experimental implementations, and writing presented in this thesis represent my original, primary contributions.

Any textual, structural, or visual similarities between this thesis and the aforementioned manuscripts are a direct result of my authorship.

Table of Contents

Acknowledgments **iv**

0.1 Disclaimer v

List of Tables **x**

List of Figures **xii**

Abstract **xviii**

Chapter 1

Introduction **1**

1.1 Motivation 1

1.2 Research Contributions 3

1.3 Thesis Organization 4

Chapter 2

Related Work **5**

2.0.1 Object Counting 5

2.0.2 Prompt Tuning 6

Chapter 3

Chain-of-Look Spatial Reasoning for Dense Surgical Instrument Counting **8**

3.1 Introduction . . . 9
3.2 Surgical Instrument Counting Dataset . . . 12
3.3 Chain-of-Look Spatial Reasoning . . . 13
3.3.1 Problem Formulation . . . 13
3.3.2 Visual Chain Generator . . . 14
3.3.3 Neighboring Loss . . . 16
3.3.4 Training . . . 18
3.3.5 Inference . . . 20
3.4 Experiments . . . 20
3.4.1 Implementation Details . . . 20
3.4.2 Evaluation Metrics . . . 20
3.4.2.1 Counting Metrics . . . 20
3.4.2.2 Localization Metrics . . . 21
3.4.3 Quantitative Results . . . 23
3.4.4 Qualitative Results . . . 26
3.4.5 Post Processing Operator . . . 27
3.4.6 Inference Speed . . . 28
3.4.7 Ablation Study . . . 29
3.4.8 Failure Analysis . . . 34
3.4.9 Analysis on Visual Chain Reasoning via Neighboring Loss 35
3.4.10 CSL Prompts Effect and Contrastive Feature Learning . . . 35
3.4.11 Divide and Conquer Inference . . . 36
3.5 Limitations of Generalization . . . 39

3.6 Chapter Summary 40

Chapter 4

Structured Object Counting with Visual Chain Reasoning **41**

4.1 Introduction 41

4.2 Method 45

4.2.1 Problem Formulation 45

4.2.2 Chain-of-Look Counting 45

4.2.2.1 Chain Constructor 45

4.2.2.2 Relative Chain Position Encoding 47

4.2.3 Module Compatibility 48

4.2.4 Loss Objective 49

4.2.5 Bi-Directional Counting 49

4.3 Experiments 50

4.3.1 Baselines and Datasets 50

4.3.2 Implementation Details 51

4.3.3 Quantitative Results 53

4.3.4 Qualitative Results 53

4.3.5 Error Analysis 55

4.3.6 Mechanistic Analysis 55

4.3.7 RPE Bias Distribution Analysis 57

4.3.8 Ablation Study 57

4.4 Limitations 58

4.5 Chapter Summary 58

Chapter 5

Future Work 63

Chapter 6

Conclusion 65

Bibliography 67

List of Tables

3.1 Comparison with state-of-the-art methods, including: (1) counting and detection methods spanning detection-based (DQ-DETR), density-based (CountGD, REC), and diffusion-based (CrowdDiff) approaches; (2) multimodality large vision-language model (Qwen-2.5-VL). 25

3.2 GAME scores (L1, L2, L3) for different methods. 26

3.3 Comparison of localization metrics results across different methods. 26

3.4 **Ablation study results.** $\Delta\mathcal{L}_{neigh}$: without Neighboring Loss; ΔCSL: without class-specific learnable prompts; Δ Visual Exemplars: without visual exemplars; ΔPost: without post processing. 30

3.5 **Prompt Placement** Performance comparison across CSL prompt placements. 31

3.6 **Prompt Initialization Strategy** Prepending task-specific initialized CSL prompts yields better performance compared to random initialization. 31

3.7 **Counting & Localization Metrics: LoRA vs. CSL Tokens.** The Mean IoU is the average IoU of all the matched bounding boxes in the test set. 34

3.8 **Gradient Magnitude Analysis** Multi-Loss scaling factor selection. 34

4.1 Threshold settings (δ_c and σ_f) across different baseline models and datasets. 52

4.2 **Quantitative Evaluation on Structural Datasets.** We compare the vanilla baselines against our CoLC module across four counting strategies: starting from the closest point (S_1), the farthest point (S_2), the image-level Ensemble, and the theoretical optimal direction (GT-Select). 54

4.3 **Error Analysis: Average Overcounts and Undercounts.** The average number of extra objects (Overcount) and missed objects (Undercount) per image across all datasets. 54

4.4 **Mean Query Peak (%) across Decoder Self-Attention Layers.** Text colors indicate intervention intensity: Low, Medium, and High. The "$\wedge$" pattern across each row illustrates how CoLC peaks in the middle layers before fading away. Analysis performed using $\texttt{CountSE+CoLC}_{S_1}$. 56

4.5 **Ablation on Position Embedding Chain Length.** Impact of the chain length (L_{chain}) on counting performance. **Bold** indicates the best performance, while underline indicates the worst. Lower-density datasets (CARPK, PUCPR+) exhibit a non-linear trend, degrading initially before recovering at the shortest chain length. Analysis performed using $\texttt{CountSE+CoLC}_{S_1}$. 56

4.6 The percentage of active queries whose maximum attention shift (peak shift) successfully targets another valid chain candidate during decoder self-attention. Results evaluate the CountSE baseline across multiple datasets. 57

List of Figures

1.1 **Accuracy vs. Generality.** Conceptual trend illustrating the inherent trade-off between model accuracy and open-set generality. Performance is high for class-specific models but declines as they generalize to open-set conditions. 2

3.1 **High-density surgical instrument counting.** Counting surgical instruments reliably in high density scenarios is challenging due to severe visual clutter and tight spatial packing of objects. To improve robustness, we propose Chain-of-Look Spatial Reasoning to introduce *visual chains* into the counting process, explicitly modeling the sequential characteristic of human visual counting. In the above figure, the first column indicates original high-density surgical instrument images, the second column presents visual chains and the third column shows the predicted counting results, where detected surgical instrument handles are highlighted with laser points. 8

3.2 (A) **Representative images from the SurgCount-HD dataset.** Sample images from the dataset, showing typical variations and an example annotation. (B) **Test result from GPT5.** We evaluate GPT-5 on an example from our SurgCount-HD dataset, where detected surgical instruments are highlighted with red dots. GPT-5 predicts a count of 84, whereas the ground truth is 57. 12

3.3 **Architecture of Chain-of-Look Spatial Reasoning framework.** High density surgical instrument images are first fed into visual chain generator to produce visual chains. Neighboring loss is further applied to guide the counting process following the visual chain. 13

3.4 CSL Prompts Initialization with BERT Text Encoder 16

3.5 **Visual Chain Generator and Neighboring loss function.** (a) Detailed architecture of Visual Chain Generator; (b) Neighboring loss and Distance loss. Detailed illustrations on the architecture can be found in Section 3.3. 17

3.6 **Qualitative results.** We present qualitative results from our CoLSR. Predicted surgical instruments number and ground-truth number are listed on each image. The detected surgical instrument handles are highlighted with laser points, which are also highlighted with red bounding boxes. 24

3.7 **Generalization ability analysis.** We evaluate our model's generalization ability via in the wild images in operating rooms. The detected surgical instrument handles are highlighted with laser points, which are also highlighted with red bounding boxes. . . . 25

3.8 **Comparison with SOTA methods.** Our CoLSR approach is compared with four existing SOTA methods for counting: CountGD, DQ-DETR, CrowdDiff and REC. For the four figures on the left side, green dots represent ground-truth, red dots represent predictions from different models. 27
3.9 Time comparison between human counting and our model 28
3.10 Example of duplicate points highlighted 28
3.11 **Ablation Studies. (a)** The highlighted region shows where the model failed to make correct predictions, indicating the model's limited ability to form coherent visual chains. **(b)** Missed handles are mostly in areas with unclear boundary separation, making them harder to detect without class-specific learnable prompts. **(c, d)** Compared with the ablated results in (a) and (b), CoLSR effectively generates accurate predictions for the location of tightly packed surgical instrument handles. 29
3.12 Predicted bounding boxes using the LoRA method. Boxes are noticeably oversized and misaligned. 32

3.13 **Analysis on Chain-of-Look Visual Reasoning via Spatial Neighboring Loss.** Left: original surgical image. Right: attention maps from different decoder layers. "$-L_{\text{neigh}}$" denotes models trained without the Neighboring Loss, whereas "$+L_{\text{neigh}}$" indicates models trained with it. The visualizations show the self-attention outputs of the Cross-Modality Decoder, where each query corresponds to one surgical instrument (indexed 0-9). Queries and their associated attention distributions are ordered left-to-right according to the instrument labels in the original image. For each setting, we display attention maps from the first decoder layer (Layer 0), which primarily captures low-level spatial relationships, and from the final decoder layer (Layer 5), which reflects higher-level semantic focus. 36

3.14 **a)** Original Input Image **b)** Image-Text Attention Map extracted from the Feature Fusion Block - Without CSL Prompts **c)** Image-Text Attention Map when trained with CSL Prompts **d)** Image-CSL Token Attention Map 37

3.15 **a)** Prediction with single-pass inference **b)** Prediction with Divide-and-Conquer approach 38

3.16 **Robust inference samples captured from multiple angles.** 40

4.1 **Overview of structured object counting.** **(A)** While existing counting frameworks treat all spatial distributions uniformly, we define **Structured Object Counting** as a distinct task subset characterized by high structural regularity. By introducing a lightweight, human-like sequential reasoning framework (CoLC), our approach enhances performance across detector-based methodologies in zero-shot text-guided scenarios. **(B), (C)** [2] Real-world demonstrations of our sequential reasoning approach, which generates a continuous visual chain (colored dashed lines) across deterministic object layouts. **(D)** In structured object counting scenarios, Gemini 3 [3] fails to adhere to a consistent, human-like sequential traversal strategy, resulting in unreliable counting outcomes. As highlighted by the yellow dashed boxes and red lines, Gemini 3 frequently deviates from the underlying spatial order, leading to omissions or double counting. 42

4.2 **CoLC Framework.** Our approach introduces two key components: (1) a Chain Constructor (CC) and (2) a Relative Chain Position Encoding (RCPE) module. The CC utilizes object queries generated by the encoder to construct a 1D sequence chain from the 2D spatial layout. Subsequently, the RCPE computes the pairwise relative positions between active chain candidates, encoding and applying them as a bias to the query attention weights in the decoder self-attention layer. All baselines share this generic GroundingDINO[4]-style architecture featuring DETR blocks, with our novel addition shaded in blue. 46

4.3 **Qualitative comparison on CARPK, PUCPR+ & SurgCount-HD datasets.** Each row displays the Ground Truth annotations (left), the Baseline predictions (middle), and our CoLC predictions (right). Regions where a model fails are highlighted with a red indicator ( ) and corresponding successful with a green indicator ( ). 60

4.4 **Qualitative comparison on SKU110K dataset.** Each row displays the Ground Truth annotations (left), the Baseline predictions (middle), and our CoLC predictions (right). Regions where a model fails are highlighted with a red indicator ( ) and corresponding successful with a green indicator ( ). 61

4.5 **Qualitative of Bi-directional Counting.** Each row displays the S2 Direction $\texttt{CoLC}_{S_2}$ (left), and the S1 Direction $\texttt{CoLC}_{S_1}$ (right) results. . 62

Abstract

Object counting is a fundamental computer vision task, yet existing methods historically treat it as an unstructured "bag-of-objects" problem. This disconnects from the sequential way humans count, hindering model performance in highly dense or repetitive environments. To address this limitation, this thesis introduces the **Chain-of-Look (CoL)** visual reasoning framework, a novel paradigm that explicitly incorporates human-inspired spatial structure and sequential traversal into the automated counting process.

First, to tackle densely packed clinical environments, the **Chain-of-Look Spatial Reasoning (CoLSR)** framework is developed for counting surgical instruments. By enforcing a structured visual chain via a novel neighboring loss function, CoLSR explicitly models spatial constraints to resolve severe occlusion and visual similarity. To facilitate rigorous evaluation, this research introduces SurgCount-HD, a comprehensive benchmark of 1,464 clinical images. Extensive experiments demonstrate that CoLSR significantly outperforms state-of-the-art counting approaches (e.g., CountGD, REC) and Multimodal Large Language Models (e.g., Qwen, ChatGPT).

To demonstrate the generalizability of this paradigm, the thesis further introduces a structured counting adapter for diverse scenes. The primary motivation for this approach is that modern vision architectures already possess robust

perceptual capabilities and inaccuracies generally arise from the absence of a systematic method to arrange the localized detections into a logical, topologically ordered sequence. This research bridges that gap by proposing a lightweight visual reasoning framework that injects a sequential geometric prior (visual chain) into a frozen base model, explicitly modeling the step-by-step aggregation of instances. Extensive experiments across diverse structured counting datasets demonstrate that this module consistently improves the accuracy of baseline architectures. Furthermore, its sequence-aware design enables bi-directional spatial cross-verification, enhancing prediction stability in cluttered scenes while making intermediate counting steps fully interpretable.

By shifting the counting paradigm from isolated perceptual detection to spatially ordered sequential reasoning, this thesis provides a highly accurate, versatile, and transparent solution for object counting in complex visual environments.

Chapter 1

Introduction

1.1 Motivation

Object counting is an innate cognitive skill that humans begin to develop from birth. While we are taught the abstract concept of numbers, we are rarely taught the physical mechanics of counting. Instead, we intuitively follow a spatial trajectory, creating a mental sequence to group objects and ensure no item is missed or counted twice. Consequently, machine-based object counting is one of the most fundamental tasks in computer vision. However, despite the rapid advancements in Artificial Intelligence, machine counting has been treated as a detection or density estimation task, ignoring the topological reasoning inherent to human cognition. Because current models approach counting as an unordered, bag-of-objects problem, they consistently struggle in dense, occluded, and highly repetitive scenes.

Inaccurate counting can have severe consequences in various high-stakes domains. In retail inventory management, counting errors lead to stockouts, overstocking, and significant financial losses. In traffic analysis, inaccurate

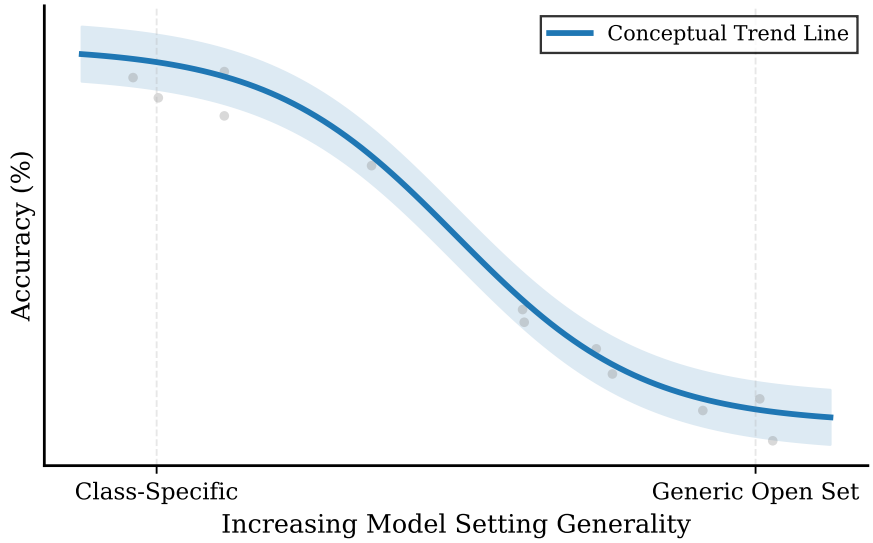

Figure 1.1: **Accuracy vs. Generality.** Conceptual trend illustrating the inherent trade-off between model accuracy and open-set generality. Performance is high for class-specific models but declines as they generalize to open-set conditions.

vehicle counts can lead to traffic congestion or safety hazards. Most critically, in surgical environments, counting errors can lead to retained surgical items inside patients, resulting in severe harm or even fatal clinical outcomes.

The current landscape of object counting models can be broadly categorized into detection-based [5, 6, 7, 8, 9, 10, 11] and density-based approaches [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22]. Density-based models focus on the local regression of object density, while detection-based models treat instances independently and aggregate them via simple summation. As illustrated in Figure 1.1, while these models can achieve high accuracy on strictly defined categories, their performance steadily declines when generalizing to open-set conditions.

To address these fundamental limitations, this thesis proposes a novel perspective on automated counting by aligning machine processes with human-like visual reasoning. Drawing inspiration from Chain-of-Thought (CoT) [23] reasoning, this work introduces **Chain-of-Look (CoL)** counting. This framework

models object counting as a sequential, spatially aware aggregation process.

1.2 Research Contributions

The core objective of this thesis is to demonstrate that inducing structural and topological biases into computer vision models significantly enhances counting accuracy, robustness, and interpretability. The key contributions are summarized as follows:

- **Novel Human-like Counting Paradigm:** This thesis identifies the limitations of current counting paradigms and demonstrates that integrating human-inspired sequential traversal (visual chains) significantly reduces omissions and duplicate counts in complex scenes.
- **CoLSR Framework for Clinical Environments:** The introduction of the Chain-of-Look Spatial Reasoning (CoLSR) framework for dense surgical instrument counting. By utilizing a novel neighboring loss and contrastive feature enhancers, CoLSR explicitly models inter-object relationships and enforces spatial constraints in highly occluded settings.
- **The SurgCount-HD Benchmark:** A novel, comprehensive dataset comprising 1,464 high-density surgical instrument images collected from diverse real-world operating rooms to facilitate rigorous evaluation of clinical counting systems.
- **CoLC Framework for Structured Scenes:** The development of Chain-of-Look Counting (CoLC), a lightweight, plug-and-play module that explicitly models sequential, spatially-ordered aggregation for generalized structured counting. CoLC introduces bi-directional coutning as a mechanism

for spatial cross-verification, providing both high accuracy and verifiable intermediate counting steps.

1.3 Thesis Organization

The remainder of this thesis is organized as follows. Chapter 2 provides a comprehensive review of the related literature. Chapter 3 addresses the challenge of counting in repetitive, highly dense environments, introducing the Chain-of-Look Spatial Reasoning (CoLSR) framework and the SurgCount-HD dataset. Chapter 4 generalizes the domain specific visual chain paradigm to diverse, structured object counting framework Chain-of-Look Counting (CoLC). Finally, Chapter 5 and 6 concludes the thesis by summarizing the core findings and outlining promising directions for future research.

Chapter 2

Related Work

2.0.1 Object Counting

Human Counting. The spatial strategies and typical scanning patterns humans employ when counting stationary objects are deeply rooted in the psychology of enumeration. Gelman et al. [24] highlight the principle of one-to-one correspondence (each item gets one and only one tag) which strongly implies a systematic way of going through a set to ensure accuracy. Logan et al. [25] further demonstrate that the number of eye movement fixations increases linearly with the number of objects during counting, indicating a sequential, item-by-item processing strategy when dealing with larger quantities.

Machine Counting. Object counting spans diverse scenarios from highly crowded scenes to unconstrained, open-world environments [14, 13, 26, 15, 27, 16, 28, 29, 30, 31, 12, 9, 10, 17, 32, 8, 7, 6, 22, 5, 20]. Methodologically, existing approaches are generally divided into density-based estimation and detection-based counting. Density-based methods [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22] regress continuous density maps and integrate them to obtain global counts, making them particularly effective in crowded scenes where precise localization is unneces-

sary. In contrast, detection-based methods [5, 6, 7, 8, 9, 10, 11] explicitly localize individual instances before aggregation, improving interpretability, verifiability, and uncertainty estimation.

Beyond architectural design, recent works enhance robustness by incorporating multimodal cues, including text prompts [10, 17, 8, 9, 5, 33], depth information [10, 9, 34, 35], 3D geometry [36], and segmentation masks [10, 9, 37]. In parallel, zero-shot and few-shot paradigms leverage learned exemplars or transferable representations [6, 38, 39, 8, 40, 41, 42], enabling competitive performance with minimal supervision and inspiring training-free counting frameworks [21, 11, 22].

More recently, researchers have explored incorporating human-inspired reasoning strategies into visual counting. Motivated by Chain-of-Thought (CoT) reasoning [23], Chain-of-Look (CoL) introduces structured visual traversal to improve compositional reasoning in vision tasks [43, 44]. Drawing a parallel to vision domain, we adopt the CoL prompting strategy to support spatial reasoning and model the sequential nature of counting by prompting the detected visual cues in a chained fashion, thereby enabling more structured visual attention across densely packed and structured objects.

2.0.2 Prompt Tuning

Parameter-Efficient Fine-Tuning (PEFT) approaches, such as prompt tuning [45], have proven effective in adapting to newer data distributions while reducing both data and computational requirements. Beyond efficiency, Yao et al. [46] demonstrate that fusing frozen tokens with learnable prompts further boosts the generalization and discriminative capability of prompt tuning. Moreover, Kang

et al. [47] introduce semantic-conditioned prompts that guide the image encoder toward extracting target-semantic-highlighted visual features.

With these existing contributions in mind, we explore integrating such PEFT approaches with the existing counting frameworks to address its limitations in handling out-of-distribution classes, particularly in highly dense scenarios.

Chapter 3

Chain-of-Look Spatial Reasoning for Dense Surgical Instrument Counting

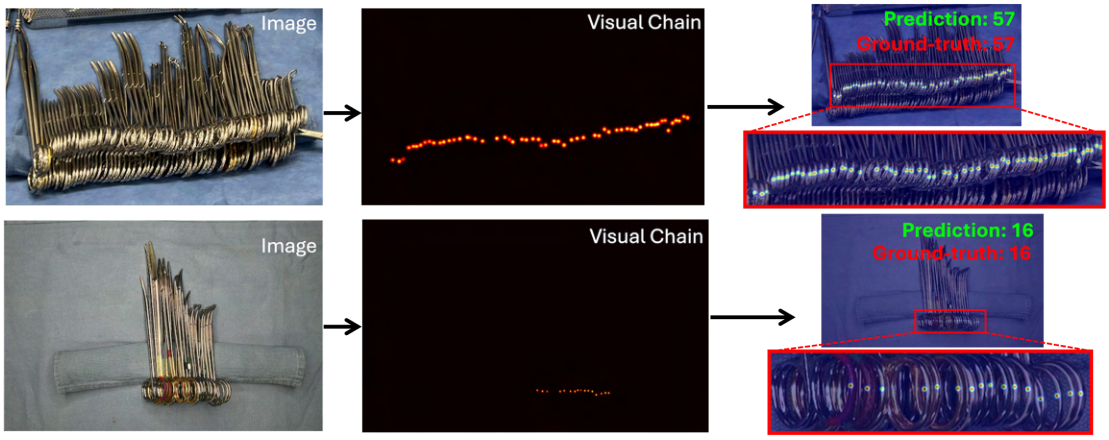

Figure 3.1: **High-density surgical instrument counting.** Counting surgical instruments reliably in high density scenarios is challenging due to severe visual clutter and tight spatial packing of objects. To improve robustness, we propose Chain-of-Look Spatial Reasoning to introduce *visual chains* into the counting process, explicitly modeling the sequential characteristic of human visual counting. In the above figure, the first column indicates original high-density surgical instrument images, the second column presents visual chains and the third column shows the predicted counting results, where detected surgical instrument handles are highlighted with laser points.

3.1 Introduction

Building upon the foundational concepts of visual reasoning discussed in previous chapters, this chapter explores its application in a high-stakes clinical environment. Counting surgical objects including instruments before and after a surgical procedure is a critical safety protocol in Operating Rooms (OR), aimed at preventing retained surgical items. Despite its importance, this process is still predominantly performed manually by surgical staff, making it time-consuming, labor-intensive, and prone to human error, particularly in high-density settings where instruments are closely clustered or visually occluded. These challenges are further exacerbated under time pressure or in emergency procedures. Given that the average operating room costs approximately $100 per minute, delays caused by manual counting can have significant financial implications in addition to clinical risks. Thus, automating the surgical instrument counting process holds great potential to reduce the workload on surgical teams, minimize human errors, and enhance workflow efficiency and patient safety. However, accurate automated counting remains a challenging task due to visual complexity and high similarity among instruments in real-world OR environments.

Most existing approaches to object counting fall into two main categories: density map-based and detection-based methods. Density map-based methods estimate object counts by summing the predicted density values across an image, while detection-based methods count the number of predicted bounding boxes. Although these methods have achieved strong performance in various scenarios such as crowd counting and open-set counting, they fundamentally treat counting as a set-based problem, ignoring the sequential nature of how humans count, particularly in complex, high-density environments. In practice,

humans typically follow a consistent visual path when counting objects to avoid omissions or duplications, and to verify the counting. For instance, as shown in Figure 3.1, technicians and nurses counting the surgical instruments tend to follow a structured visual sequence, such as scanning from left to right or right to left. This sequential reasoning process is crucial in ensuring accurate counts under cluttered and visually challenging conditions, yet it is largely overlooked in existing automated counting frameworks.

Motivated by the importance of sequential visual reasoning in human counting behavior, this chapter introduces the **Chain-of-Look Spatial Reasoning** (CoLSR) framework that explicitly models the counting sequence in dense object scenes by locating each object as a counting point. CoLSR *explicitly models the sequential nature of human counting*, which is particularly critical in *complex, high-density environments*. Introducing a direction for counting is especially important in high-stakes surgical scenarios, where medical staffs always follow a **strict counting direction** to ensure accuracy rather than counting instruments in a random order. Unlike traditional methods that treat object instances independently, our CoLSR not only predicts the locations of target objects (e.g., the handles of surgical instruments in our task) but also captures the *spatial dependencies* and *structural relationships* among them. To achieve this, we first generate visual chains using a transformer-based counting model, CountGD [8]. These visual chains provide guidance for our model, enabling it to reason spatially and improve prediction accuracy. To further align the predicted visual chains with the underlying spatial structure of the scene, we introduce a novel neighboring loss that encourages the predicted object order to match the ground-truth sequence. The neighboring loss encourages the model to consider the proximity and ordering of adjacent objects, and enforces consistency with realistic spatial

arrangements by penalizing implausible gaps or overlaps. Therefore, the neighboring loss guides the model to learn a coherent spatial chain that mirrors the sequential patterns humans naturally follow during counting, leading to more robust and accurate performance in high-density scenarios.

We evaluate our CoLSR framework through extensive experiments on a high-density surgical instrument dataset that we construct. Empirical results demonstrate that CoLSR consistently outperforms state-of-the-art (SOTA) object counting methods and multimodality large language models in the context of densely packed surgical instruments, highlighting its effectiveness in real-world, high-complexity scenarios.

The specific contributions of this chapter are:

- The introduction of the novel and challenging task of dense surgical instrument counting, a problem with significant clinical implications.
- The development of the Chain-of-Look Spatial Reasoning (CoLSR) framework which incorporates *visual chains* into the counting process, explicitly modeling the sequential nature of human visual counting. This includes the design of a novel neighboring loss to equip the model with spatial reasoning capabilities by enforcing inter-object relationships and realistic spatial constraints.
- The construction of a comprehensive dataset comprising 1,464 high-density surgical instrument images collected from diverse real-world clinical settings. Extensive experiments show that CoLSR delivers significant improvements over existing methods for high-density surgical instrument counting, achieving both high accuracy and fast inference.

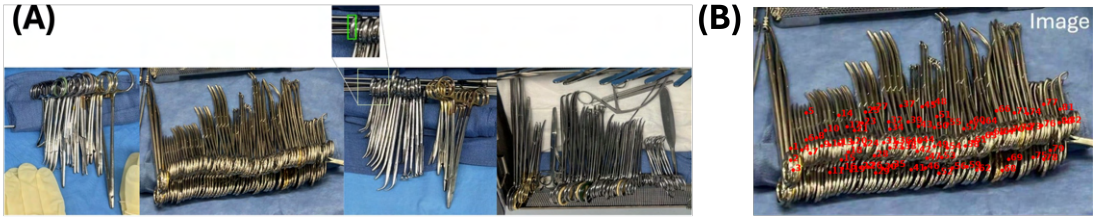

Figure 3.2: (A) **Representative images from the SurgCount-HD dataset.** Sample images from the dataset, showing typical variations and an example annotation. (B) **Test result from GPT5.** We evaluate GPT-5 on an example from our SurgCount-HD dataset, where detected surgical instruments are highlighted with red dots. GPT-5 predicts a count of 84, whereas the ground truth is 57.

3.2 Surgical Instrument Counting Dataset

We introduce SurgCount-HD, a novel dataset consisting of **H**igh-**D**ensity arrangements of **Surg**ical instruments collected prior to surgical procedures. Each image contains various types of surgical instruments compactly organized on the back table (a common surgical preparation surface). The dataset focuses on instrument layouts where handles are oriented toward the camera, and bounding-box annotations are provided for these handles, as shown in Figure 3.2 (A). Translational and rotational augmentations were applied and the final the dataset comprises 1,236 training images and 228 test images. All images were resized such that the shorter edge is scaled to 800 pixels while preserving the original aspect ratio. All annotations represent a single class, namely "*circular instrument handle*".

We used Roboflow [48] platform to manually label instrument handles across densely packed scenes. The annotation process required substantial manual effort due to the high density and visual similarity among instruments. The data collection and annotation process spanned several months and involved multiple domain experts to ensure accuracy and consistency.

The SurgCount-HD dataset presents significant challenges due to the tightly

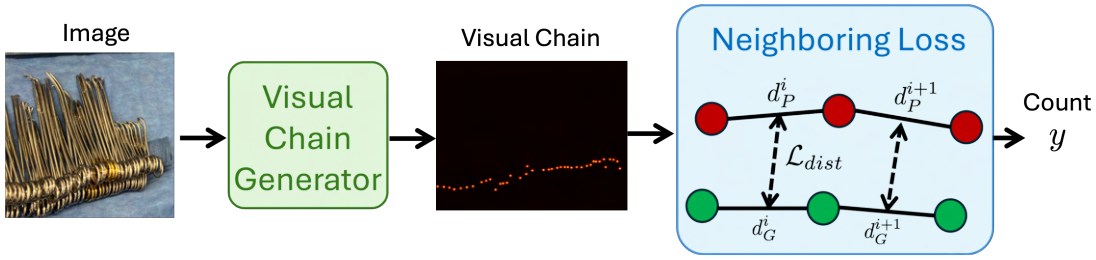

Figure 3.3: **Architecture of Chain-of-Look Spatial Reasoning framework.** High density surgical instrument images are first fed into visual chain generator to produce visual chains. Neighboring loss is further applied to guide the counting process following the visual chain.

clustered and visually occluded surgical instruments. Even for human annotators, counting in such high-density scenarios is time-consuming and labor-intensive. To assess the difficulty of this dataset, we evaluated GPT5 [49] on selected examples from SurgCount-HD. As shown in Figure 3.2 (B), GPT5 performs poorly in these dense settings (detected 84 instruments, where the ground-truth is 57), highlighting the challenge of this SurgCount-HD dataset.

3.3 Chain-of-Look Spatial Reasoning

3.3.1 Problem Formulation

In this section, we introduce the Chain-of-Look Spatial Reasoning (CoLSR) framework for high density surgical instrument counting. Given an image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ of high density surgical instruments, our goal is to train a model $\mathcal{F}_\theta$ to localize surgical instrument handles and count the number y of surgical instruments in the image based on the localized instrument handles: $y = \mathcal{F}_\theta(I)$, where θ denotes the parameters of our model, H and W represents the height and weight of the given image. Two major components construct the CoLSR framework: (1) *Visual Chain Generator* for producing the visual chain to guide the counting

process; (2) *Neighboring Loss Function* to introduce physical constraints along the visual chain.

3.3.2 Visual Chain Generator

As shown in Figure 3.3, the first part of CoLSR is to generate visual chain of the given image. The visual chain serves as a *structured visual sequence* to guide the model's counting process under cluttered and visually challenging conditions.

The visual chain generator is constructed based on the CountGD model [8]. Different from CountGD, we also take class-specific text tokens as input to enhance the quality of generated visual chain. Figure 3.5 (a) depicts the detailed architecture of Visual Chain Generator.

Image Encoder. We first encode the input image $\mathbf{I}$ with a Swin-B version of Swin Transformer [50] based Image Encoder f_I into spatial feature maps at three different scales, followed by 1×1 convolution to produce image tokens $\mathbf{z_I}$ of 256 dimensions. The visual exemplar tokens $\mathbf{z_B}$ are obtained from the image tokens using aligned region-of-interest pooling (RoIAlign) with the pixel coordinates specified by the visual exemplars $\mathbf{B}$. The generated visual exemplar tokens also have 256 dimension, which is the same with image tokens and text tokens.

Text Encoder. For text input, a BERT-based text transformer [51] encoder f_T is employed to encode the text description $\mathbf{T_S}$ into a sequence of tokens $\mathbf{z_T}$ with at most 256 dimensions. Then the n image tokens, p visual exemplar tokens and q text tokens are applied with the feature enhancer f_ϕ.

Feature Enhancer. The generated visual exemplar tokens $\mathbf{z_B}$ are fused with the text tokens $\mathbf{z_T}$ through the feature enhancer f_ϕ with 6 blocks self-attention modules. The generated fused feature $\mathbf{z_{B,T}}$ is further fused with the image

tokens $\mathbf{z_I}$ through the feature enhancer f_ϕ with 6 blocks cross-attention modules. To enhance the grounding ability of our model, we take the prompt tuning approach to introduce class-specific text tokens $\mathbf{T_C}$ as additional inputs for the feature enhancer. These class-specific text tokens serve as learnable parameters to further improve the results of generated visual chains. Therefore, the outputs from the feature enhancer are computed as:

$$\begin{aligned}\mathbf{z_{B,T}}, \mathbf{z_I} = & f_\phi((f_\theta(\mathbf{X}), \\ & \text{RoIAlign}(f_\theta(\mathbf{X}), \mathbf{B}), f_T(\mathbf{T_S}), f_T(\mathbf{T_C})).\end{aligned} \tag{3.1}$$

To implement this prompt design, the modified feature enhancer takes two different Class Specific Learnable (CSL) token instances, both initialized with the same text but diversified with Gaussian noise (Figure 3.4). The first set of tokens is prepended to the concatenated set of visual exemplar and text tokens. We treat the CSL tokens as tunable text prompts, hence they are prepended to the latent tokens derived from text. These tokens, along with the image token, are used in the bidirectional attention module, where image-text and text-image cross-attention are computed.

The second set of CSL tokens is prepended to the output of the bidirectional module and fed into the self-attention layer, where the self-attention between the text tokens is calculated. We introduce these two sets of tokens to represent distinct functional roles: one captures the relation between text and image, while the other addresses text-specific nuances. The fused feature embedding $\mathbf{F}_{encoder}$ is constructed by concatenating these components:

$$\mathbf{F}_{encoder} = [\mathbf{T}_{\text{CSL}}\,;\,\mathbf{T}_{\text{text}}\,;\,\mathbf{T}_{\text{vis}}] \in \mathbb{R}^{(l+2h)\times d}, \tag{3.2}$$

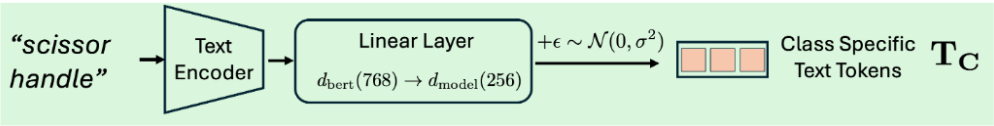

Figure 3.4: CSL Prompts Initialization with BERT Text Encoder

where $\mathbf{T}_{\text{CSL}} \in \mathbb{R}^{l \times d}$ denotes the l CSL tokens, while $\mathbf{T}_{\text{text}}$ and $\mathbf{T}_{\text{vis}}$ (both $\in \mathbb{R}^{h \times d}$) represent the text and visual exemplar tokens, respectively. Following standard prompt-tuning methodology, we discard $\mathbf{T}_{\text{CSL}}$ before passing the sequence to the decoder:

$$\mathbf{F}_{decoder} = [\mathbf{T}_{\text{text}}\,;\, \mathbf{T}_{\text{vis}}] \in \mathbb{R}^{(l+h) \times d}. \tag{3.3}$$

Query Selection. k image patch tokens are selected which achieve the highest similarity with the fused visual exemplar $\mathbf{B}$ and text description $\mathbf{T_S}$. Following CountGD, we set k to 900, serving as cross-modality queries input to the cross-modality decoder f_ψ.

Cross-modality Decoder. The cross-modality decoder f_ψ contains 6 blocks of self-attention and cross-attention to enhance the cross-modality queries. The final output of confidence score $\hat{\mathbf{Y}}$ is computed as

$$\hat{\mathbf{Y}} = \text{Sigmoid}(f_\psi(\mathbf{z_I}, \mathbf{z_{B,T}}, f_S(\mathbf{z_I}, \mathbf{z_{B,T}}^T, k))\mathbf{z_{B,T}}^T), \tag{3.4}$$

where f_S denotes the above Query Selection module.

3.3.3 Neighboring Loss

The generated visual chain provides a coarse estimate of the instrument count and serves as a structural guidance to guide our model toward precise surgical instrument counting. To incorporate directional consistency along the visual

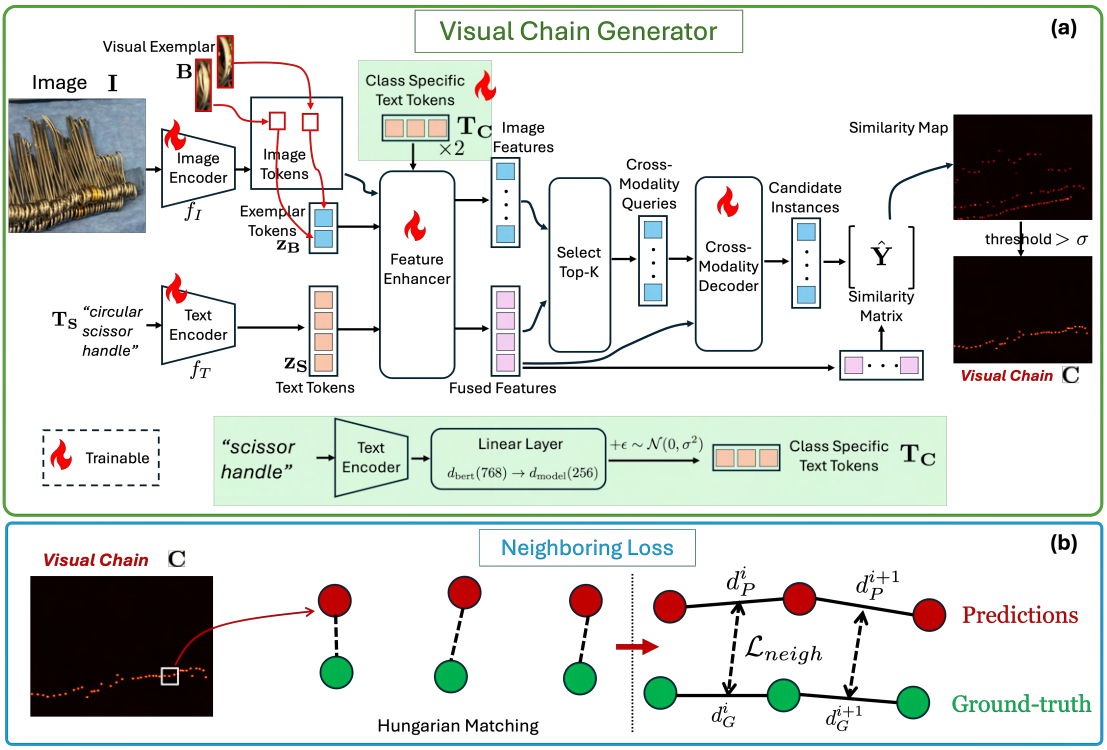

Figure 3.5: **Visual Chain Generator and Neighboring loss function.** (a) Detailed architecture of Visual Chain Generator; (b) Neighboring loss and Distance loss. Detailed illustrations on the architecture can be found in Section 3.3.

chain, we introduce a neighboring loss term into the training objective. As illustrated in Figure 3.5 (b), given the predicted visual chain **C**, we first use the Hungarian matching algorithm to associate each predicted bounding box with its corresponding ground-truth bounding box. Specifically, for the predicted $\{b_i\}_{i=1}^{N_P}$ and ground-truth surgical instrument handle $\{b_j\}_{j=1}^{N_G}$, the value function $\mathbf{v}$ for Hungarian matching algorithm is defined as:

$$\mathbf{v}_{i,j} = d_{i,j} + \quad \mathcal{L}_{i,j}^{cls}, \tag{3.5}$$

where $\mathbf{v}_{i,j}$ is the value function for the pair (i, j) from predictions and ground-truth, $d_{i,j}$ indicates the l_1 norm of the center points and $\mathcal{L}_{cls}$ denotes the classifi-

cation cost (see Equation 3.8 for further details).

Given the matched bounding boxes, we examine the local regions of detected surgical instrument handles in a fixed direction (either left-to-right or right-to-left). As illustrated in Figure 3.5 (b), we introduce a neighboring loss that encourages the distances between adjacent center points of bounding boxes in the predictions to closely match those in the ground truth:

$$\mathcal{L}_{neigh} = \sum_{i=1}^{N} ||d_P^i - d_G^i||_2, \tag{3.6}$$

where d_P^i denotes the distance between two neighboring center points of predicted bounding boxes, d_G^i indicates the distance between two counterpart neighboring center points of ground-truth bounding boxes. This neighboring loss function promotes spatial consistency in the ordering of instruments and enforces a visual chain structure in the model's reasoning process, enabling the Chain-of-Look mechanism. We further discuss this effect in Section 3.4.9.

3.3.4 Training

As shown in Figure 3.5 (a), we train the image encoder f_I, text encoder f_T, feature enhancer f_ϕ, cross-modality decoder f_ψ and the learnable class specific text tokens $\mathbf{T_C}$. The optimization objective of the whole model includes CountGD [8]'s original bounding box localization loss, classification loss, and our proposed neighboring loss:

$$\begin{aligned}\mathcal{L} &= \lambda_{loc}\mathcal{L}_{loc} + \lambda_{neigh}\mathcal{L}_{neigh} + \lambda_{cls}\mathcal{L}_{cls} \\ &= \lambda_{loc}\sum_{i=1}^{N_G}|\hat{c}_i - c_i| + \lambda_{neigh}\sum_{i=1}^{N_G}||d_P^i - d_G^i||_2 \\ &+ \lambda_{cls}FocalLoss(\hat{\mathbf{Y}}, T),\end{aligned} \tag{3.7}$$

where λ_{loc}, λ_{cls} and λ_{neigh} are weights to control each loss term, $\hat{\mathbf{Y}}$ is the similarity matrix from Equation 3.4 and $T \in \{0,1\}^{N_P \times (N_G+1)}$ denotes the optimal Hungarian matching between the N_P predicted queries and the N_G ground truth handle instances, including an additional label for "no object" similar to CountGD.

To establish this optimal matching T, we utilize the CountGD formulation to define the Hungarian matching value function $v(i,k)$ between prediction i and ground-truth k. Given $\alpha = 0.25, \gamma = 2$:

$$\begin{aligned}v(i,k) &= \underbrace{\|\mathbf{b}_i - \mathbf{b}_k\|_1}_{\textbf{bbox cost}} + \underbrace{\sum_{j=1}^{C}\tilde{y}_{kj}\left[\mathcal{L}_{\text{pos}}(p_{ij}) - \mathcal{L}_{\text{neg}}(p_{ij})\right]}_{\textbf{cls cost}}, \\ \text{s.t.} \quad \mathcal{L}_{\text{pos}}(p) &= -\alpha(1-p)^{\gamma}\log(p+\varepsilon), \\ \mathcal{L}_{\text{neg}}(p) &= -(1-\alpha)p^{\gamma}\log(1-p+\varepsilon),\end{aligned} \tag{3.8}$$

During training, the model receives a high density surgical instrument image $\mathbf{I}$ along with visual exemplars $\mathbf{B}$ as inputs. These inputs are processed through the image encoder and cross-modality modules to generate query representations, which are then optimized using the aforementioned loss functions.

3.3.5 Inference

During inference, we only pass a high density surgical instrument image $\mathbf{I}$ as input to our model. The outputs are predicted surgical instrument handle bounding boxes. We further execute a post processing operator $\mathcal{P}$ to remove the redundant predicted bounding boxes that share the horizontal regions more than a predetermined threshold τ. Detailed descriptions of post processing operator can be found in section 3.4.5.

3.4 Experiments

3.4.1 Implementation Details

We train the model for 30 epochs with a learning rate of 1×10^{-4} using the Adam optimizer and a weight decay of 1×10^{-4}, which is reduced by a factor of ten after the 10^{th} epoch. Training is performed with a batch size of 4 on a single NVIDIA RTX 3090 GPU. The multi-loss weights are set as follows: $\lambda_{\text{loc}} = 10$, $\lambda_{\text{neigh}} = 100$, and $\lambda_{\text{cls}} = 1$. The number of CSL prompts used is 64, and the confidence threshold σ is set to 0.26. The rest of the training setup, including data pre-processing and augmentation strategies, follows the original CountGD [8] configuration.

3.4.2 Evaluation Metrics

3.4.2.1 Counting Metrics

MAE, RMSE. We use the standard Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) as evaluation metrics.

$$\begin{aligned} \text{MAE} &= \frac{1}{N}\sum_{i=1}^{N}|N_P - N_G|, \\ \text{RMSE} &= \sqrt{\frac{1}{N}\sum_{i=1}^{N}(N_P - N_G)^2}, \end{aligned} \tag{3.9}$$

where N is the number of image samples, N_P is the predicted count and N_G is the ground truth count for image N_i.

Grid Average Mean Absolute Error. We also measure the Grid Average Mean Absolute Error (GAME) [52] to evaluate the spatial accuracy of the predicted counts within each image. GAME quantifies how well the counting predictions are localized across subdivided regions of the image. Moreover, we also use detection-related counting metrics such as precision, recall and F1-score as defined in Equation. 3.10.

3.4.2.2 Localization Metrics

Since the number of predicted instrument locations may not match the ground truth (GT) annotations, computing localization accuracy is non-trivial. To address this, we first filter predictions by selecting only those whose center points fall within any GT bounding box. These filtered predictions are then matched to GT points. In cases where multiple predictions fall within the same GT box, we apply the Hungarian algorithm using L2 distance as the cost function to perform one-to-one matching.

Unmatched predictions are treated as missed detections, while matched pairs are used to compute localization metrics. Specifically, for each image, we calculate the mean L2 distance (average localization error), the median L2 distance (typical

error at the 50th percentile), and the 95th percentile of L2 distances (representing the worst 5% of matched localizations). To obtain a single dataset-level metric, we take the mean of these three values across all images. A similar procedure is applied for computing the Mean IoU reported in the Table. 3.7.

Steps for a single input :

$$
\begin{aligned}
&P_{\text{filtered}} = p \in P_{\text{pred}} \mid \exists b \in B_{GT} \text{ such that } p \in b \\
&M^* = \underset{M}{\operatorname{argmin}} \sum_{(p,g)\in M} |p - g|_2 \\
&d_i = |p_i - g_i|_2 \\
&\bar{d} = \frac{1}{N} \sum_{i=1}^{N} d_i \\
&\text{Median Error} = \text{median}(d_1, d_2, \ldots, d_N) \\
&\text{95th Percentile Error} = P_{95}(d_1, d_2, \ldots, d_N) \\
&\text{True Positive (TP)} = N \\
&\text{False Positive} = len(P_{\text{pred}}) - TP \\
&\text{False Negative} = len(G_{GT}) - TP
\end{aligned}
\tag{3.10}
$$

where :

P_{pred} : The set of all predicted center points.

B_{GT} : The set of all ground truth (GT) bounding boxes.

G_{GT} : The set of all GT center points.

P_{filtered} : The set of all filtered center point prediction.

M^* : The optimal one-to-one matching

d_i : The L2 distance for the i-th matched pair(p_i, g_i)

N : The total number of matched pairs.

3.4.3 Quantitative Results

MAE, RMSE. In Table 3.1, we compare the performance of CoLSR with state-of-the-art (SOTA) methods on the task of high-density surgical instrument counting. For a fair comparison, all SOTA counting baselines are finetuned on our SurgCount-HD dataset, except Qwen. CoLSR outperforms all competing methods in both MAE and RMSE metrics. The major reason lies in the primary limitation of existing counting methods, where they treat object instances as independent entities, lacking the spatial reasoning necessary to capture the dependencies and structural relationships among densely packed instruments. In contrast, CoLSR explicitly models physical constraints, enabling it to reason over spatial arrangements and structural coherence more effectively. In addition, multimodality large vision-language models (MLVL) such as GPT5 [49] and Qwen-2.5-VL [53] also perform much worse compared with CoLSR, where the MAE of MLVLs are more than 10 times higher than CoLSR.

GAME Score. Given that surgical instruments in our dataset are typically con-

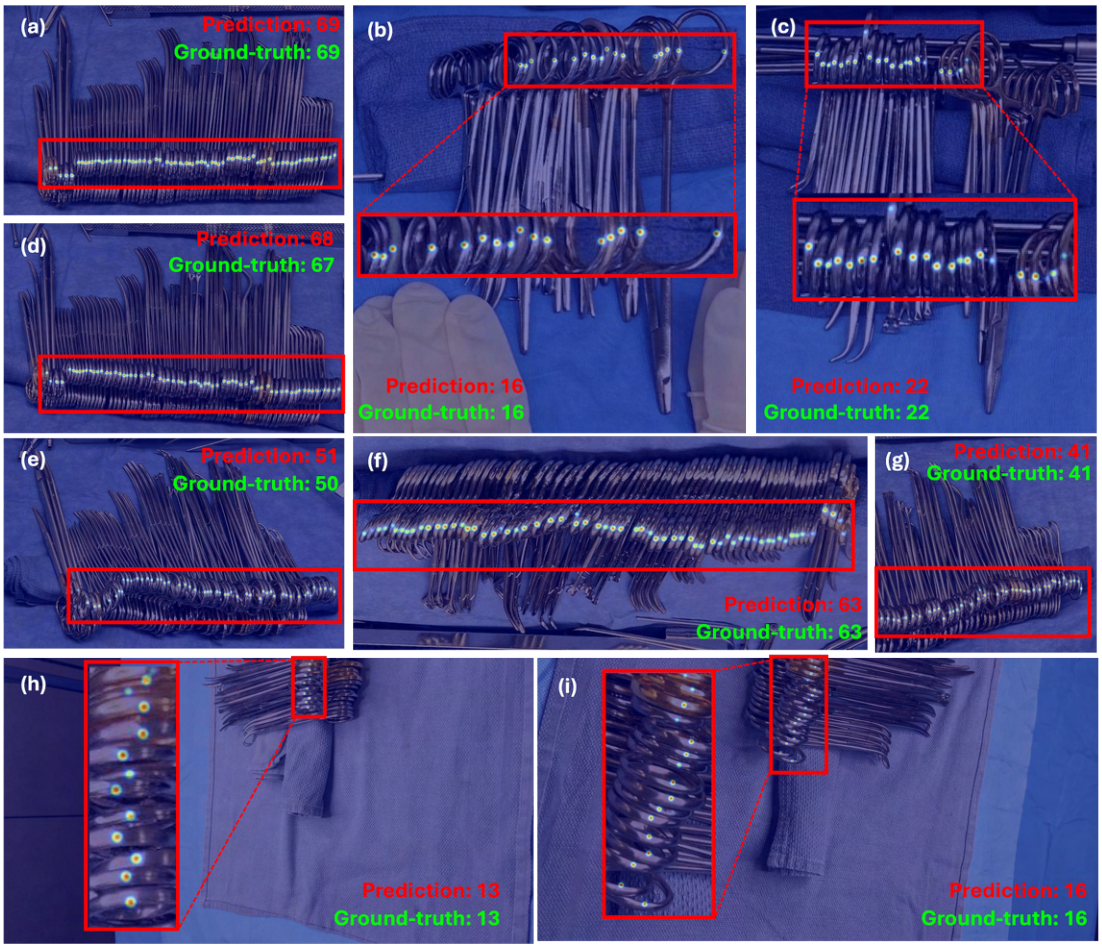

Figure 3.6: **Qualitative results.** We present qualitative results from our CoLSR. Predicted surgical instruments number and ground-truth number are listed on each image. The detected surgical instrument handles are highlighted with laser points, which are also highlighted with red bounding boxes.

centrated within a limited spatial area, the GAME scores tend to decrease as the grid resolution parameter L increases (Table. 3.2). This is due to the presence of numerous grids containing no instruments, which contribute zero error to the overall score. Of the 228 images in our test set, only 98 include instance-level annotations suitable for spatial evaluation. Therefore, the GAME scores and localization metrics were calculated exclusively on this subset.

Localization Metrics. Table 3.3 presents the localization performance of CoLSR

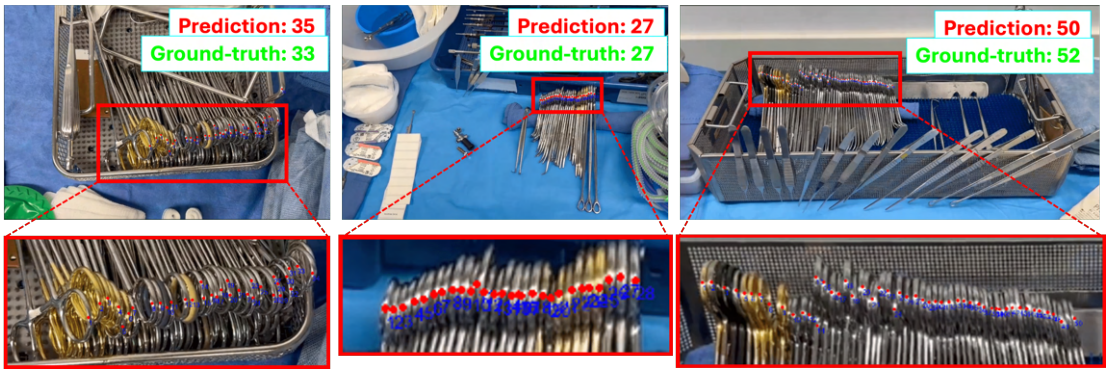

Figure 3.7: **Generalization ability analysis.** We evaluate our model's generalization ability via in the wild images in operating rooms. The detected surgical instrument handles are highlighted with laser points, which are also highlighted with red bounding boxes.

Method	MAE ↓	RMSE ↓
CountGD [8]	7.84	10.71
DQ-DETR [54]	4.24	6.81
CrowdDiff [19]	18.63	22.93
REC [5]	2.82	4.50
Qwen2.5-VL-7B-Instruct [53]	17.06	21.72
CoLSR (Ours)	**0.88**	**1.27**

Table 3.1: Comparison with state-of-the-art methods, including: (1) counting and detection methods spanning detection-based (DQ-DETR), density-based (CountGD, REC), and diffusion-based (CrowdDiff) approaches; (2) multimodality large vision-language model (Qwen-2.5-VL).

compared to other methods. Our approach achieves the best results across all metrics, including Mean L2 distance, Mean of Median L2 distance, Mean of 95th-Percentile L2 distance, Precision, Recall, and F1 score. These results demonstrate that CoLSR not only accurately counts the instruments but also precisely localizes them, which is crucial for applications requiring precise spatial understanding of the scene.

Method	GAME-L1 ↓	GAME-L2 ↓	GAME-L3 ↓
CountGD [8]	1.01	0.41	0.14
REC [5]	0.60	0.25	0.08
DQ-DETR [54]	0.68	0.25	0.07
CoLSR (Ours)	**0.54**	**0.23**	**0.07**

Table 3.2: GAME scores (L1, L2, L3) for different methods.

	CountGD [8]	REC [5]	DQ-DETR [54]	**CoLSR (Ours)**
Mean L2 distance ↓	12.79	6.89	**5.84**	6.43
Mean of Median L2 distance ↓	12.01	6.33	**5.46**	5.99
Mean of 95th-Percentile L2 distance ↓	21.05	12.66	**10.56**	11.44
Precision ↑	0.41	0.73	0.84	**0.85**
Recall ↑	0.41	0.74	0.81	**0.84**
F1 score ↑	0.41	0.74	0.83	**0.85**

Table 3.3: Comparison of localization metrics results across different methods.

3.4.4 Qualitative Results

Figure 3.6 presents qualitative results of high-density surgical instrument counting using CoLSR. The visualizations highlight the robustness of our approach across various challenging scenarios, including variations in camera angles (Figure 3.6 (f), (h), (i)), instrument orientations (Figure 3.6 (b), (c), (e)), and dense packing patterns (Figure 3.6 (a), (d)).

Figure 3.8 provides a visual comparison between our approach and existing SOTA methods for high-density instrument counting. As shown, SOTA methods often fail to detect all instrument handles, particularly in cluttered regions, resulting in under-counting. In contrast, CoLSR accurately localizes the instrument handles, as indicated by the cropped bounding boxes, demonstrating its effectiveness in handling densely packed scenes. To evaluate the generalization ability of our method, we test our model on in the wild images from operating rooms. Results in Figure 3.7, 3.16 indicate our method continually achieves robust results, demonstrating the generalization ability to real world operating

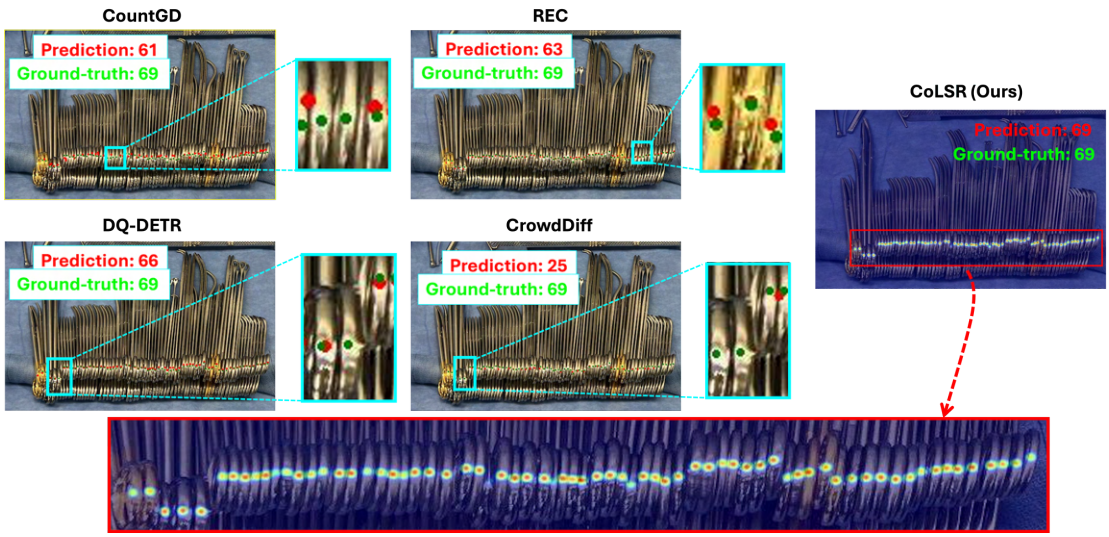

Figure 3.8: **Comparison with SOTA methods.** Our CoLSR approach is compared with four existing SOTA methods for counting: CountGD, DQ-DETR, CrowdDiff and REC. For the four figures on the left side, green dots represent ground-truth, red dots represent predictions from different models.

room scenarios.

3.4.5 Post Processing Operator

Due to the dense and ambiguous appearance of the instruments in the images, the model frequently produces multiple duplicate detections close to each other (Fig. 3.10). To mitigate this, we applied a post-processing step to eliminate such points.

First, we sort the detected center points from left to right or top to bottom based on their orientation (Equation 3.11). For each detected point, we examine neighboring points within a distance threshold θ along the given axis. If multiple points are found within this range, we retain only the point with the highest confidence score and discard the others, Algorithm 1.

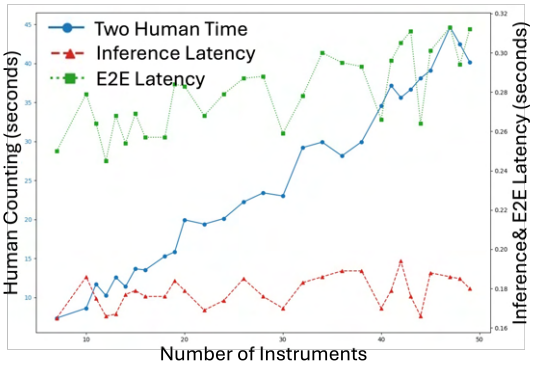

Figure 3.9: Time comparison between human counting and our model

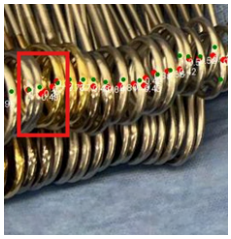

Figure 3.10: Example of duplicate points highlighted

Algorithm 1 Point Selection with Distance Threshold

for $P_i, P_j \in \{left, right\}$ **do**
  if $|P_i - P_j| < d$ **then**
    $P_{selected} \leftarrow \arg\max_{P \in \{P_i, P_j\}} \text{conf}(P)$
    $P_{removed} \leftarrow \arg\min_{P \in \{P_i, P_j\}} \text{conf}(P)$
    Remove $P_{removed}$ from set
  end if
end for

3.4.6 Inference Speed

Our model is lightweight and achieves fast inference, running over ***100*** $\times$ ***faster*** than manual human counting. Our mobile application achieves a peak end-to-end (E2E) latency (including pre-processing, inference, and post-processing) of only 0.32s, compared to 44s required for manual counting. Average latency is 0.28 ± 0.02s for our mobile application versus 25.12 ± 11.63s for human counting. The experiments were performed across a range of scenarios, with the number of surgical instruments varying from 7 to 49 per trial. In each trial, two individuals performed manual counts, followed by a count using the mobile application. Figure 3.9 illustrates the contrast in performance between the traditional method and the app-based approach, highlighting the real-time efficiency gains enabled

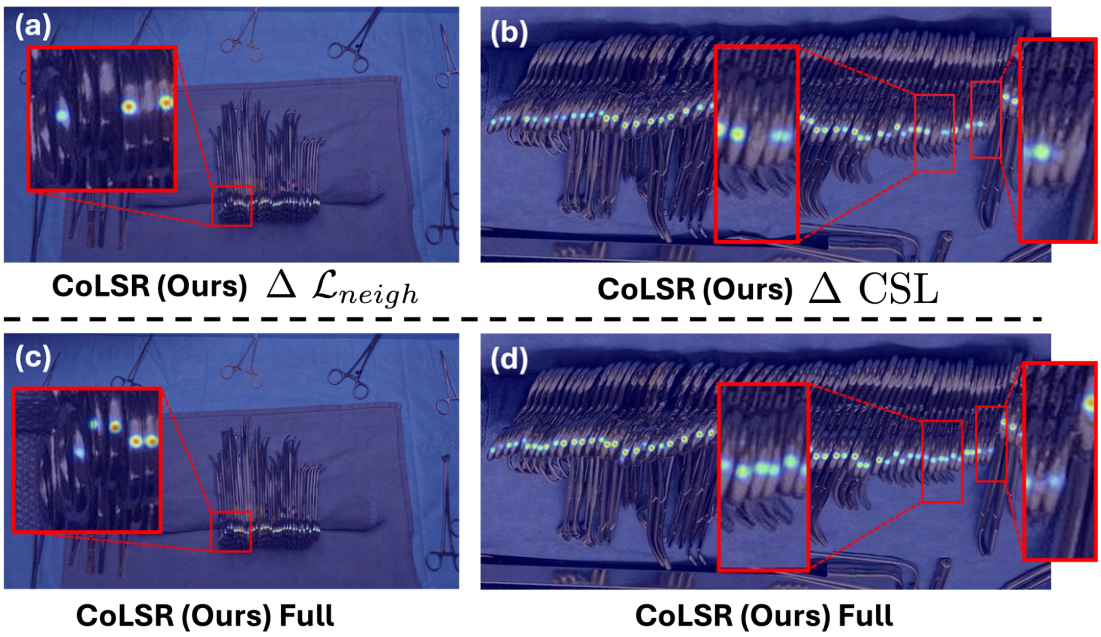

Figure 3.11: **Ablation Studies. (a)** The highlighted region shows where the model failed to make correct predictions, indicating the model's limited ability to form coherent visual chains. **(b)** Missed handles are mostly in areas with unclear boundary separation, making them harder to detect without class-specific learnable prompts. **(c, d)** Compared with the ablated results in (a) and (b), CoLSR effectively generates accurate predictions for the location of tightly packed surgical instrument handles.

by the proposed system.

3.4.7 Ablation Study

We conduct the following ablation studies to verify the effectiveness of each proposed component, including the neighboring loss, the class-specific learnable prompts and visual exemplars.

Effectiveness of neighboring loss function ($\Delta\mathcal{L}_{neigh}$). Removing the neighboring loss diminishes the model's ability to construct a visual chain for spatial reasoning, as shown in Fig.3.11 (a). This leads to a performance drop of approximately 105% in terms of MAE, as shown in Table 3.4.

Method	MAE ↓	RMSE ↓
$\Delta\mathcal{L}_{neigh}$	1.81	2.73
ΔCSL	2.05	3.30
ΔVisual Exemplars	1.5	2.21
ΔPost	0.996	1.48
CoLSR (Full)	**0.88**	**1.27**

Table 3.4: **Ablation study results.** $\Delta\mathcal{L}_{neigh}$: without Neighboring Loss; ΔCSL: without class-specific learnable prompts; Δ Visual Exemplars: without visual exemplars; ΔPost: without post processing.

Role of Learnable CSL Tokens (Δ CSL). Eliminating the CSL prompts significantly impairs the model's ability to detect fine-grained handle boundaries (further discussed in section 3.4.10). This is further exacerbated when instruments are densely packed, causing the handle boundaries to appear merged as highlighted in Figure. 3.11 (b). Consequently, the model's performance degrades by approximately 133% in terms of MAE, as shown in Table 3.4.

Pure Zero-shot training and inference (Δ Visual Exemplars). As shown in Table 3.4, training and evaluation in a purely zero-shot setting without visual exemplars leads to a performance drop of approximately 70% in terms of MAE.

Role of postprocessing (Δ Post). We remove postprocessing during inference and found slight drop of both MAE and RMSE in Table 3.4.

CSL Prompts Placement: Appending vs. Prepending Previous studies [55] have highlighted how prompt placement affects transformer models. Our analysis (Table 3.5) reveals that prompt placement significantly impacts performance, with prepending yielding 32% better MAE than appending. To understand this performance disparity, we analyzed two key metrics using a consistent training batch for both configurations:

- **CSL Token Gradient Norms**: We measured the gradient norms for both

Placement	MAE ↓	RMSE ↓	CSL Token Grad ↑	Vision Attention Weights ↑
Append	1.30	1.98	Total: 0.044 Avg: 0.0037	Mean: 0.00225 Std: 0.00325
Prepend	0.88	1.27	Total: 7.876 Avg: 0.656	Mean: 0.00643 Std: 0.00498

Table 3.5: **Prompt Placement** Performance comparison across CSL prompt placements.

Initialization Type	MAE	RMSE
Random	1.32	1.96
Semantic (”scissor handle”)	0.88	1.96

Table 3.6: **Prompt Initialization Strategy** Prepending task-specific initialized CSL prompts yields better performance compared to random initialization.

text and fusion CSL tokens across all six encoder layers. Averaging these over the layers assesses their relative contribution during backpropagation, indicating that prepended prompts provide on average 177× stronger supervision signals.

- **Vision Multi-Head Attention Weights at Fusion Module**: To analyze the influence on visual attention, we extracted prompt-related weights from the fusion module. Prompt-related weights were stacked and averaged across all four attention heads, followed by averaging over the batch dimension. The same process was repeated for each of the six encoder layers, and the resulting layer-wise averages were further averaged to obtain a final mean value. This analysis reveals that prepended prompts maintain 3× stronger coupling with image features.

We further investigate the impact of prompt initialization when prompts are prepended. Table 3.6 shows that using task-specific initialization leads to notable performance improvements.

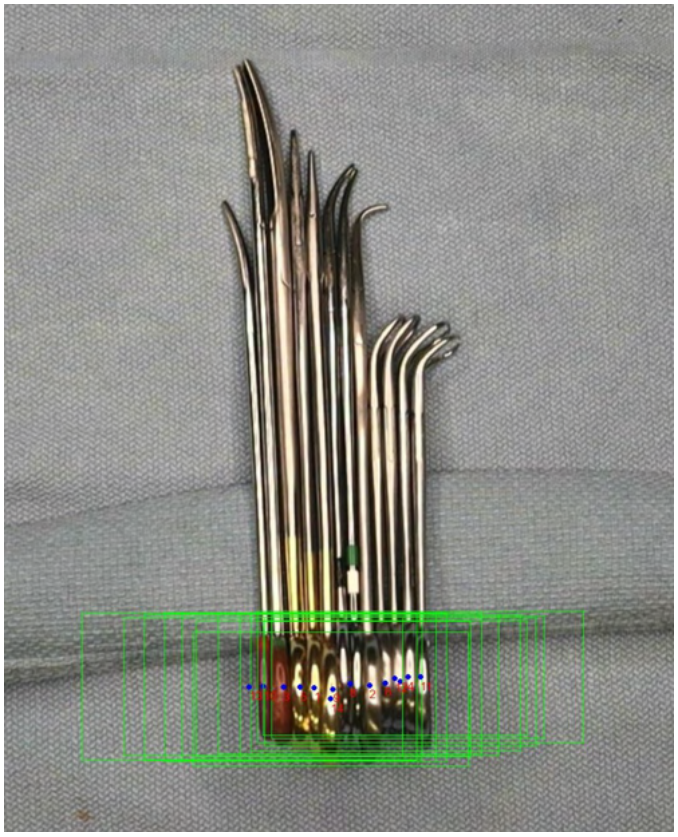

Figure 3.12: Predicted bounding boxes using the LoRA method. Boxes are noticeably oversized and misaligned.

LoRA versus CSL Tokens. We explore whether adding explicit spatial conditioning using learnable prompt tokens offers benefits over weight adaptation methods in our instrument counting task. To test this, we inserted LoRA [56] adapters at the fusion and text encoder layers, mirroring the placement of CSL tokens. The adapter configuration (α=32, Rank =16) was chosen to match the parameter count of the CSL tokens, allowing for a fair comparison.

LoRA Parameters

- Rank=16, α=32
- LoRA Params Per Layer : 16 (Rank) x 256 (In Features) + 16 (Rank) x 256

(Out Features) = 8,192 parameters

- Text Encoder layers: 6 layers x 2 LoRA modules x 8,192 Per Layer $\approx$ 98K parameters
- Fusion layers: 6 layers x 2 LoRA modules x 8,192 $\approx$ 98K parameters
- Total LoRA Parameters $\approx$ 196K parameters

CSL Parameters

- Text Encoder layers: 64 (CSL Token) x 256 (feature dim) x 6 layers $\approx$ 98K parameters
- Fusion layers: 64 (CSL Token) x 256 (feature dim) x 6 layers $\approx$ 98K parameters
- Total CSL Parameters $\approx$ 196K parameters

Our comparison between CSL Tokens and LoRA shows that token-level spatial conditioning leads to superior object detection performance despite LoRA's parameter efficiency. In our experiments, LoRA struggled with instrument localization, often missing center points and producing inaccurate bounding boxes (Figure. 3.12). This suggests that LoRA's weight-space adaptation may lack the direct spatial conditioning beneficial for precise object localization in our setting. In contrast, the contrastive learning capability shown with CSL tokens (Section 3.4.10) appears to improve spatial reasoning that goes beyond parameter efficiency considerations. Our findings suggest that for this spatially sensitive detection task, explicit spatial conditioning through prompt tokens may provide capabilities that our constrained low-rank weight modification approach could not achieve.

Method	MAE ↓	RMSE ↓	Mean L2 (matched) ↓	Mean IoU (matched) ↑
LoRA	5.63	7.66	10.42	0.028
CSL Tokens	0.88	1.27	6.38	0.290

Table 3.7: **Counting & Localization Metrics: LoRA vs. CSL Tokens.** The Mean IoU is the average IoU of all the matched bounding boxes in the test set.

$(\lambda_{cls}, \lambda_{loc}, \lambda_{neigh})$	$\mathcal{L}_{cls}$ Grad	$\mathcal{L}_{loc}$ Grad	$\mathcal{L}_{neigh}$ Grad	MAE	RMSE
(1, 1, 1)	0.915	0.0005	0.0002	3.23	4.20
(1, 10, 10)	0.737	0.0007	0.0003	2.35	3.59
(1, 10, 100)	**28.50**	**0.055**	**0.106**	**0.88**	**1.27**

Table 3.8: **Gradient Magnitude Analysis** Multi-Loss scaling factor selection.

Multi-Loss Weight Selection. Our method incorporates three distinct loss functions: Cross-Entropy Loss ($\mathcal{L}_{cls}$), Distance Loss ($\mathcal{L}_{loc}$), and Neighboring Loss ($\mathcal{L}_{neigh}$). We measured gradient norms across key shared model layers (encoder, decoder, fusion, text) for three loss weighting configurations to validate our λ selection strategy.

As shown in Table 3.8, there exists a high imbalance in gradient magnitudes between the cross-entropy (CE) loss and auxiliary losses, with the latter exhibiting gradients 1,000-4,000x weaker under equal weighting. As a result, auxiliary objectives are effectively ignored during training. By introducing a loss weighting configuration of λ = (1, 10, 100), we observe a substantial increase in auxiliary contribution (0.56% vs. 0.09%) while preserving CE dominance. This leads to a 30x increase in total gradient activity, enabling more expressive multi-objective optimization.

3.4.8 Failure Analysis

Most errors arise from the dense and visually ambiguous appearance of instruments in the images. Empirical results indicate that performance degrades in

scenarios where gaps or occlusions disrupt the continuity of surgical instruments, making spatial reasoning more challenging. Potential solutions include leveraging multi-view inputs (e.g., short video sequences capturing multiple viewpoints) or incorporating depth information to better handle severe occlusions.

3.4.9 Analysis on Visual Chain Reasoning via Neighboring Loss

We demonstrate that the neighboring loss enforces a Chain-of-Look mechanism within the model's reasoning process. Figure 3.13 visualizes the self-attention scores of the query proposals in the Cross-Modality Decoder. At the first decoder layer (Layer 0), where the model primarily captures low-level spatial cues, we observe that removing the neighboring loss results in higher attention entropy, with focus spread across non-adjacent queries. In contrast, applying $\mathcal{L}_{\text{neigh}}$ constrains each query to attend mainly to its immediate predecessors and successors, forming a snake-like chained structure.

At the final decoder layer (Layer 5), the attention maps show that this chained behavior also shapes high-level semantic reasoning. For instruments that are densely clustered (labels 5-8), the model leverages the most visible and confident queries as anchors, reflected by the pronounced dark attention band, to improve the representation of uncertain and ambiguous queries. These structured interactions suppress hallucinations and ultimately improve counting accuracy.

3.4.10 CSL Prompts Effect and Contrastive Feature Learning

When trained without CSL prompts, the model's attention is spread across and less focused on the handle regions, as illustrated in Fig. 3.14(b-c). In contrast, CSL tokens learn contrastive features, as shown in Fig. 3.14(d), where the attention

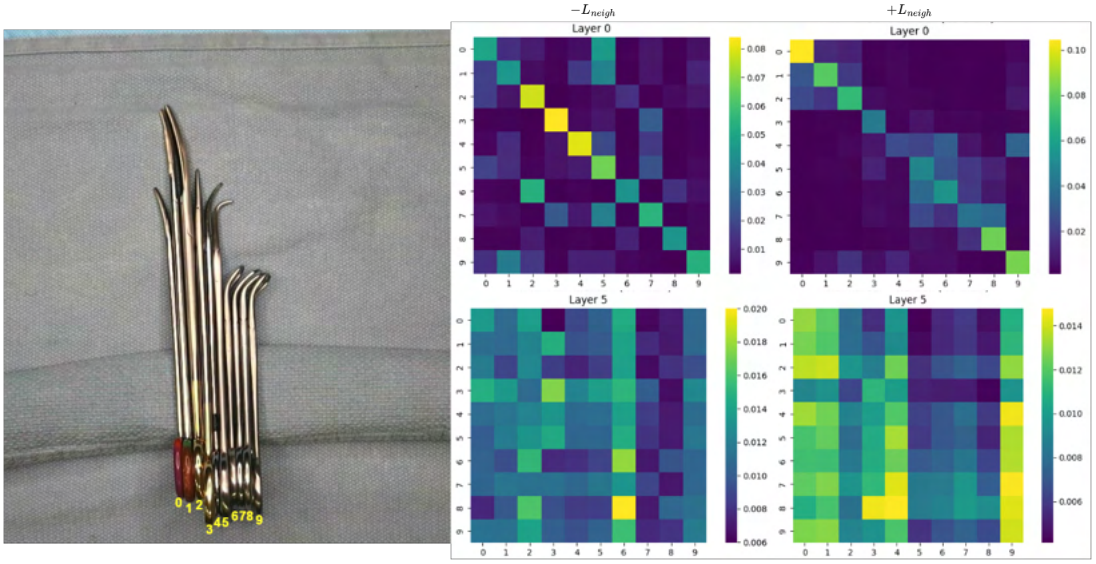

Figure 3.13: **Analysis on Chain-of-Look Visual Reasoning via Spatial Neighboring Loss.** Left: original surgical image. Right: attention maps from different decoder layers. "$-L_{\text{neigh}}$" denotes models trained without the Neighboring Loss, whereas "$+L_{\text{neigh}}$" indicates models trained with it. The visualizations show the self-attention outputs of the Cross-Modality Decoder, where each query corresponds to one surgical instrument (indexed 0-9). Queries and their associated attention distributions are ordered left-to-right according to the instrument labels in the original image. For each setting, we display attention maps from the first decoder layer (Layer 0), which primarily captures low-level spatial relationships, and from the final decoder layer (Layer 5), which reflects higher-level semantic focus.

on the handle is minimal. This complementary negation helps the text tokens to attend to the handle regions more precisely. We experimented with varying numbers of CSL prompts {16, 32, 64, 128}, and found that 64 prompts produced the best performance based on the MAE metric.

3.4.11 Divide and Conquer Inference

CoLSR is designed to handle densely packed instrument clusters, which are the most common setup in real-world surgeries. However, its performance degrades

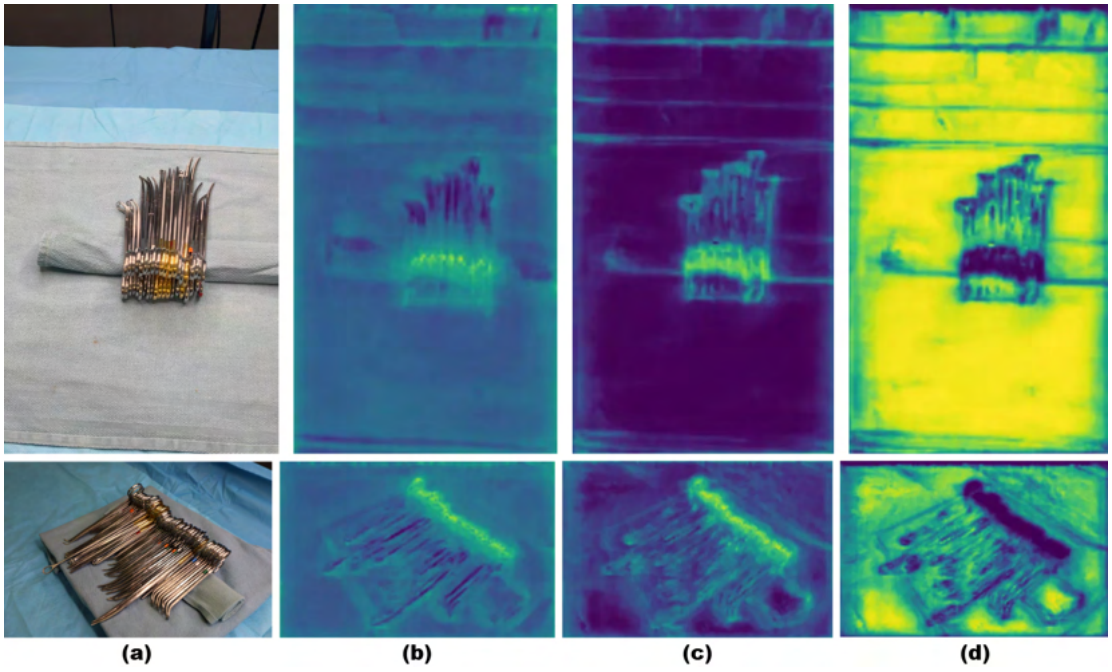

Figure 3.14: **a)** Original Input Image **b)** Image-Text Attention Map extracted from the Feature Fusion Block - Without CSL Prompts **c)** Image-Text Attention Map when trained with CSL Prompts **d)** Image-CSL Token Attention Map

when multiple dense clusters are spatially separated (Figure 3.15a). This is due to the visual chain constraint enforced by the neighboring loss, which fails to capture long-range dependencies across large gaps. To address this, we follow a two-stage approach: the Divide-and-Conquer strategy (Algorithms 2 and 3).

In the first stage, the entire image is processed by the network to obtain an initial set of predicted bounding boxes. To determine the sequence for distance checking, we must first establish the dominant orientation of the instruments. We extract the center points from these initial predictions and compute the difference between the maximum and minimum coordinates along the x- and y-axes. The

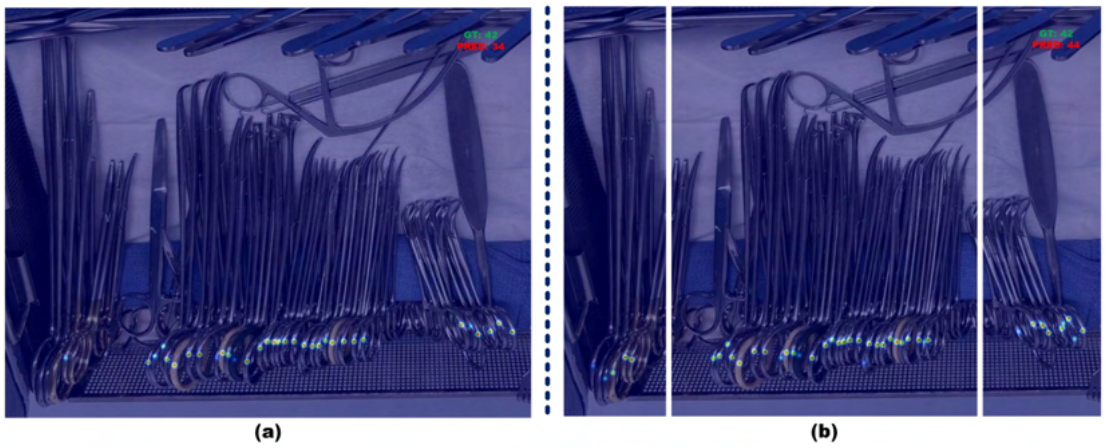

Figure 3.15: **a)** Prediction with single-pass inference **b)** Prediction with Divide-and-Conquer approach

axis with the largest spatial span is considered the dominant orientation:

$$
\begin{aligned}
&1 - \operatorname{int}\left(\left(P_x^{\max} - P_x^{\min}\right) > \left(P_y^{\max} - P_y^{\min}\right)\right) \\
&\quad = \begin{cases} 0 & \text{for } x\text{-axis} \\ 1 & \text{for } y\text{-axis,} \end{cases}
\end{aligned} \tag{3.11}
$$

where the predicted coordinate sets are defined as:

$$
\begin{aligned}
P_{\text{pred}} &= \{(x_i, y_i) \mid \text{point } i \text{ is predicted}\}, \\
P_x &= \{x_i \mid (x_i, y_i) \in P_{\text{pred}}\} \quad \text{with } P_x \subseteq [0, W], \\
P_y &= \{y_i \mid (x_i, y_i) \in P_{\text{pred}}\} \quad \text{with } P_y \subseteq [0, H],
\end{aligned} \tag{3.12}
$$

and W and H denote the width and height of the image, respectively.

Once the points are sorted along this dominant axis, we compute the L_2 norm between adjacent center points. If the distance between two neighbors exceeds a predefined threshold δ, the sequence is split: points on the left are grouped into one cluster, and those on the right begin another. This distance-based clustering

Algorithm 2: Distance-Based Clustering

```
clusters ← {}
cluster ← [0]                                   ▷ cluster start
for i = 0 to |pred_points| − 2 do
    p_i ← pred_points[i]
    p_{i+1} ← pred_points[i + 1]
    if ‖p_i − p_{i+1}‖_2 > δ then
        cluster.append(i)                       ▷ cluster end
        clusters.append(cluster)
        cluster ← [i + 1]                       ▷ next cluster start
    end if
end for
slices ← slice_image(clusters)
return slices
```

Algorithm 3: Two-Pass Counting

```
pred_points ← run_inference(image)              ▷ first pass
slices ← create_cluster(pred_points, δ)
final_detections ← {}
for slice ∈ slices do
    pred_points ← run_inference(slice)          ▷ second pass
    final_detections.append(pred_points)
end for
return final_detections
```

is repeated until all center points are assigned to spatially isolated clusters.

Each identified cluster is then cropped from the original image, and second-stage inference is performed independently on each localized dense region. Finally, the predictions of all the clusters are stitched back together to produce the final output (Figure 3.15b).

3.5 Limitations of Generalization

While our method demonstrates robustness to variations in angle and lighting conditions typical of operating room (OR) environments (Figure. 3.16, 3.7), the scope of this work is limited to surgical instrument counting, as indicated by the paper title. Consequently, generalization to other domains may require further investigation.

Figure 3.16: **Robust inference samples captured from multiple angles.**

3.6 Chapter Summary

We introduce the Chain-of-Look spatial reasoning framework that is inspired by human sequential counting behavior, designed to improve accuracy in densely packed surgical instrument scenes. By enforcing a structured visual chain and introducing a neighboring loss to model spatial constraints, our method outperforms existing state-of-the-art counting models as well as multimodality large language models. This framework offers a generalizable approach that can be extended to broader applications requiring spatial reasoning in dense visual environments. Furthermore, this chapter presented SurgCount-HD, a high-density surgical instrument dataset constructed to facilitate benchmarking and drive future research in this domain. While this framework offers a highly effective approach for uniform, dense visual environments, extending spatial reasoning to diverse, structured objects presents new challenges. The following chapter addresses these challenges by exploring structured object counting with visual chain reasoning.

Chapter 4

Structured Object Counting with Visual Chain Reasoning

4.1 Introduction

While the previous chapter demonstrated the effectiveness of spatial reasoning in densely packed, uniform environments, extending these capabilities to diverse, structured scenes presents a distinct set of challenges. Object counting is a fundamental task in computer vision with wide applications ranging from retail inventory to industrial inspection and traffic analysis. The general problem of object counting has received considerable attention, where density-based and detection-based approaches have achieved remarkable progress.

However, a practically important and underexplored subclass called structured object counting remains insufficiently addressed by existing methods. Many real-world counting scenarios exhibit strong spatial regularities. In structured environments such as grocery shelves or parking lots, objects are arranged in repeated, ordered layouts. Unlike unstructured scenes where objects appear

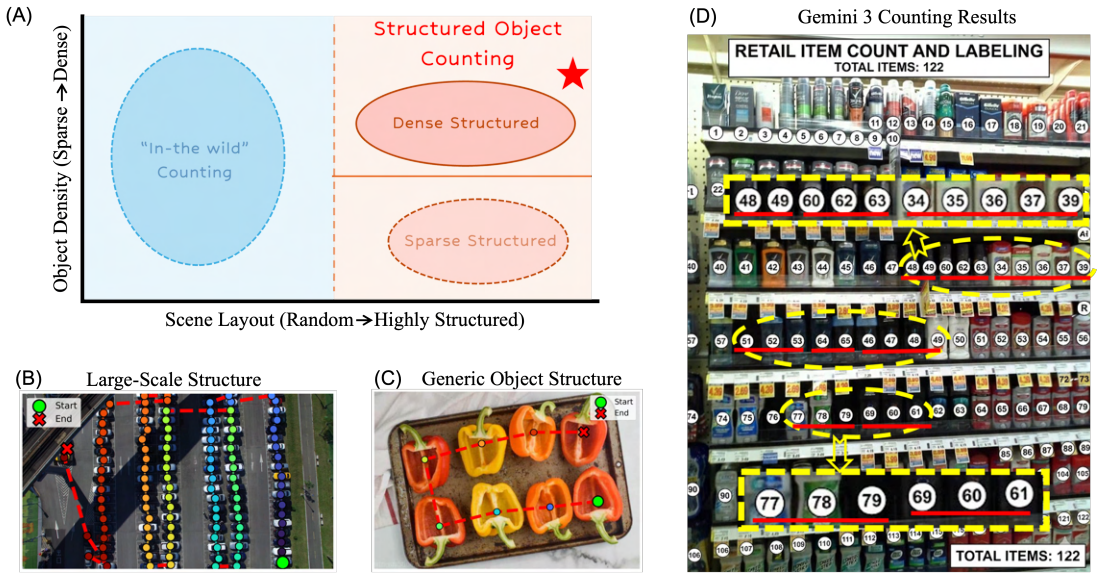

Figure 4.1: **Overview of structured object counting. (A)** While existing counting frameworks treat all spatial distributions uniformly, we define **Structured Object Counting** as a distinct task subset characterized by high structural regularity. By introducing a lightweight, human-like sequential reasoning framework (CoLC), our approach enhances performance across detector-based methodologies in zero-shot text-guided scenarios. **(B), (C)** [2] Real-world demonstrations of our sequential reasoning approach, which generates a continuous visual chain (colored dashed lines) across deterministic object layouts. **(D)** In structured object counting scenarios, Gemini 3 [3] fails to adhere to a consistent, human-like sequential traversal strategy, resulting in unreliable counting outcomes. As highlighted by the yellow dashed boxes and red lines, Gemini 3 frequently deviates from the underlying spatial order, leading to omissions or double counting.

irregularly and without consistent spatial organization, structured scenes admit natural sequential traversal strategies (Fig. 4.1-A, B, C). Humans typically count such scenes by following rows or columns in a consistent direction, ensuring that each instance is visited exactly once. This structured traversal reduces double-counting and omissions, leading to reliable results even in visually cluttered conditions.

Despite this observation, existing object counting methods largely ignore spatial structure. Density-based models focus on local regression of object density,

while detection-based models treat instances independently and aggregate them via simple summation. Although effective in general settings, these approaches lack an explicit mechanism to organize detections into a coherent, spatially ordered counting procedure. As a result, errors in structured scenes often arise not from perceptual failures, where modern detectors are already highly capable, but from the absence of a principled aggregation strategy that respects the underlying object layout. As shown in Fig. 4.1-D, even the state-of-the-art multimodal reasoning model Gemini 3 [3] fails to adhere to a consistent, human-like sequential traversal strategy, resulting in unreliable counting outcomes.

In this chapter, we introduce Chain-of-Look Counting (CoLC), a visual reasoning framework designed to explicitly incorporate spatial structure into the counting process. Our central insight is that counting in structured scenes should be modeled as a **sequential aggregation process** rather than a purely local detection or density estimation task. Instead of redesigning existing counting architectures, CoLC operates as a lightweight Counting that can be attached to any base counting model. Given detected instances or intermediate representations, CoLC constructs an ordered "*visual chain*" that follows the structural layout of the scene and models step-by-step instance aggregation along this chain. This design transforms structured counting from unordered accumulation into a spatially organized reasoning process.

CoLC offers several key advantages. **(I)** It is plug-and-play. It requires no architectural modification to the underlying counting network and introduces minimal computational overhead. **(II)** Its sequence-aware formulation naturally supports bi-directional traversal, enabling the model to count in multiple spatial directions and cross-verify predictions. This cross-checking mechanism improves robustness in cluttered or ambiguous scenes where layout regularity

may be partially occluded. **(III)** By explicitly modeling intermediate aggregation steps, CoLC provides transparent and interpretable counting trajectories, offering insight into how the final count is produced.

We evaluate CoLC on multiple structured counting benchmarks spanning diverse object categories and scene types. Across different baseline architectures, CoLC consistently improves counting accuracy, demonstrating its effectiveness and generality. These results highlight the importance of incorporating spatial structure into counting and suggest that reasoning over detection sequences is a promising direction for structured visual understanding. The specific contributions of this chapter are:

- This chapter identifies and formalizes the problem of structured object counting, highlighting the failure of existing methods to exploit spatial regularity.
- The introduction of CoLC, a lightweight plug-and-play structured counting framework that explicitly models sequential, spatially-ordered aggregation of detections for structured counting. CoLC incorporates bi-directional counting as a mechanism for spatial cross-verification, improving prediction stability in cluttered or ambiguous scenes.
- Extensive experiments demonstrate that CoLC achieves consistent performance improvements across multiple baselines and structured counting benchmarks and provides reliable intermediate counting steps.

4.2 Method

4.2.1 Problem Formulation

We define structured object counting as a distinct subset within the object counting space (Fig. 4.1-A) where objects $\{o_i\}_{i=1}^{N}$ exhibit deterministic, repeated layouts (Fig. 4.1-B, C). To validate this, we target the challenging zero-shot text-guided setting across multiple baselines, demonstrating our module's broad compatibility and ease of integration. Crucially, by showing performance improvements while keeping the base models completely frozen, we demonstrate that existing counting-based vision architectures have an inherent affinity for sequential reasoning. This suggests that the ability to perform structured, stepwise counting is not something that must be learned from scratch.

4.2.2 Chain-of-Look Counting

We introduce the Chain-of-Look Counting (CoLC) framework to mimic human-like visual sequential reasoning within existing detector-based counting frameworks. CoLC consists of two core components: a heuristic-based Chain Constructor (CC) and a Relative Chain Position Encoding (RCPE) module (Fig. 4.2).

4.2.2.1 Chain Constructor

The encoder (or the Feature Enhancer in models such as CountGD [8] and CountSE [6]) outputs query proposals consisting of confidence logits q_i^{logit} and reference center points q_i^{center} in pixel space. We first filter these proposals using a confidence threshold δ_c to isolate valid chain candidates. Queries falling below this threshold are assigned a default chain position of 0. In particular, assigning

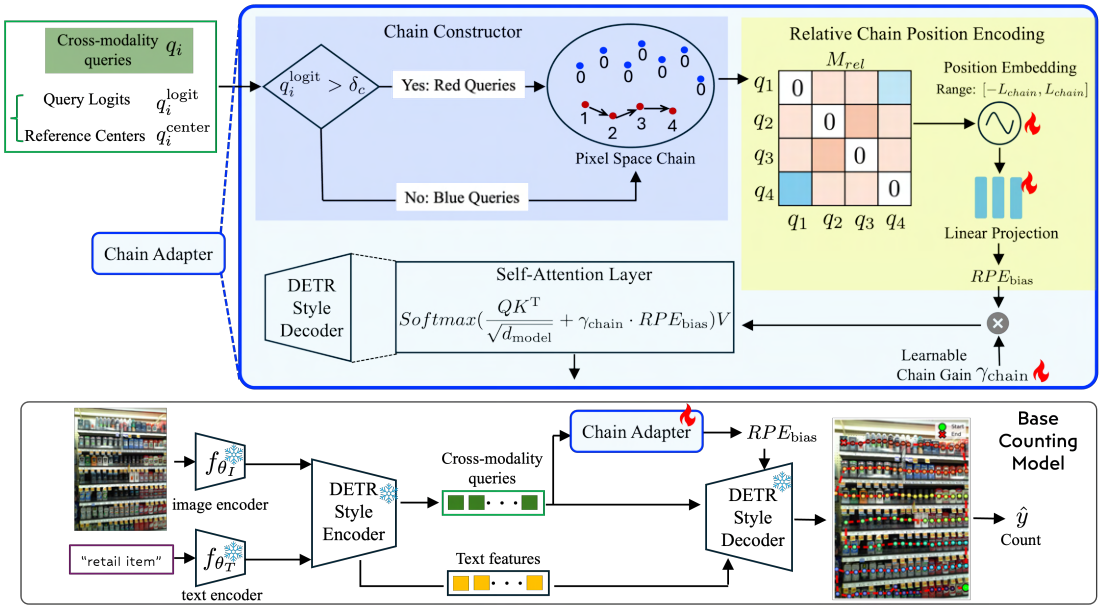

Figure 4.2: **CoLC Framework.** Our approach introduces two key components: (1) a Chain Constructor (CC) and (2) a Relative Chain Position Encoding (RCPE) module. The CC utilizes object queries generated by the encoder to construct a 1D sequence chain from the 2D spatial layout. Subsequently, the RCPE computes the pairwise relative positions between active chain candidates, encoding and applying them as a bias to the query attention weights in the decoder self-attention layer. All baselines share this generic GroundingDINO[4]-style architecture featuring DETR blocks, with our novel addition shaded in blue.

a 0 position to all queries serves as the default behavior when the chain module is bypassed entirely. For the remaining high-confidence queries, we construct a Hamiltonian path $\mathcal{P}$ to establish a sequential order, as illustrated in Alg 2.

While the starting point of this path can be user-defined, we default to the coordinate closest to the origin (i.e., the minimum squared L_2 norm). From this origin, Algorithm 2 iteratively builds the path by greedily moving to the nearest unvisited neighbor based on spatial Euclidean distance. Once the sequence π is formed, we map this ordered path back into a positional map P. Each active query in the chain is assigned an increasing integer index (1 to $|\pi|$), while queries below the confidence threshold δ_c remain as background 0. Finally, this

Algorithm 2 Visual Chain Construction

Input: Query logits q^{logit}, Reference centers q^{center}, Threshold δ_c
Output: Absolute position map P (where 0 indicates background)

$P \leftarrow \mathbf{0}$ ▷ Initialize all positions to background
$V \leftarrow \{i \mid q_i^{logit} > \delta_c\}$ ▷ Filter active queries
if $V = \emptyset$ **then**
    return P
end if
$curr \leftarrow \arg\min_{i \in V} \|q_i^{center}\|_2^2$ ▷ Start at top-leftmost node
$\pi \leftarrow [curr]$
Visited $\leftarrow \{curr\}$
while $|\text{Visited}| < |V|$ **do**
    $next \leftarrow \arg\min_{j \in V \setminus \text{Visited}} \|q_{curr}^{center} - q_j^{center}\|_2$ ▷ Nearest neighbor
    Append $next$ to π
    Visited $\leftarrow$ Visited $\cup \{next\}$
    $curr \leftarrow next$
end while
for $k = 1$ **to** $|\pi|$ **do**
    $P[\pi_k] \leftarrow k$ ▷ Assign topological sequence mapping
end for
return P

process converts the 2D spatial layout of the objects into a straightforward 1D Hamiltonian path $\mathcal{P}$, serving as the structural input to the relative encoding module.

4.2.2.2 Relative Chain Position Encoding

The chain sequence generated from the Chain Constructor is then utilized to construct a pairwise relative position matrix M_{rel}. M_{rel} is then passed through a learnable position embedding of dimension d_{model}. To limit the embedding space, we clip the relative positions to a predefined range $[-L_{chain}, L_{chain}]$, restricting the distinct positional embeddings to this $2L_{chain}$ interval. As the decoder relies on Multi-Head Attention (MHA) blocks, these embeddings are processed

through a multi-head linear projection resulting in RPE_{bias}. Finally, the projected relative position embeddings are scaled by a learnable chain gain parameter γ_{chain} to ensure their magnitude aligns with the original attention scores. These scaled embeddings are then added directly to the self-attention matrix prior to the Softmax operation, following the relative position encoding formulation introduced in [57].

$$\text{SelfAttention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_{model}}} + \gamma_{chain} \cdot RPE_{bias}\right) V. \tag{4.1}$$

This architecture design allows the sequential chain to be toggled without disrupting the original architecture. During training, the base model remains entirely frozen, and only the Chain Constructor is optimized. As a result, our module is highly parameter-efficient, introducing an overhead of only $N_{dec} \times (d_{model} \cdot n_{heads} + 1) + (2L_{chain} + 1) \cdot d_{model}$ trainable parameters, where N_{dec} denotes the number of decoder layers and n_{heads} is the number of attention heads.

4.2.3 Module Compatibility

While our current implementation of the chain module is specifically designed for DETR-based architectures (e.g., GroundingDINO), the underlying methodology generalizes to other architectures suiting sequential reasoning, given that appropriate structural adaptations are made.

4.2.4 Loss Objective

To demonstrate the broad applicability of our method, we integrate it into multiple baselines (CountGD[8], CountSE[6], and CoLSR[1]) without modifying their underlying objective functions. This ensures that the base loss functions remain unchanged during training, with gradients backpropagated only to update the Chain Counting parameters, preserving the original optimization dynamics of the chosen baseline.

For CountGD and CountSE, the loss $\mathcal{L}$ combines an L_1 localization loss ($\mathcal{L}_{loc}$) on object centers and a focal classification loss ($\mathcal{L}_{cls}$):

$$\mathcal{L} = \lambda_{loc}\mathcal{L}_{loc} + \lambda_{cls}\mathcal{L}_{cls} = \lambda_{loc}\sum_{i=1}^{l}|\hat{c}_i - c_i| + \lambda_{cls}\text{FocalLoss}(\hat{\mathbf{Y}}, T), \tag{4.2}$$

where $\hat{c}_i$ and c_i are the predicted and ground-truth centers, T represents the optimal Hungarian matching target, and λ denotes the trade-off weights.

For the CoLSR baseline, the objective incorporates an additional neighborhood reasoning term ($\mathcal{L}_{neigh}$):

$$\mathcal{L} = \lambda_{loc}\sum_{i=1}^{N_G}|\hat{c}_i - c_i| + \lambda_{neigh}\sum_{i=1}^{N_G}\|d_P^i - d_G^i\|_2 + \lambda_{cls}\text{FocalLoss}(\hat{\mathbf{Y}}, T), \tag{4.3}$$

where d_P^i and d_G^i are the predicted and ground-truth distances, respectively, for the N_G ground truth instances.

4.2.5 Bi-Directional Counting

To mimic the human way of verifying a count by recounting from a different starting direction, we propose bi-directional counting. Since our greedy path

construction is deterministic, changing the starting point generates a distinct visual chain. The new path modifies the Relative Position Encoding (RPE_{bias}), shifting how object queries interact within the self-attention layer contextually. We leverage this multi-viewpoint verification by evaluating two starting points: the reference point closest to (S_1) and farthest from (S_2) the image origin. To produce a more robust final count, we report metrics for both directions alongside an image-level ensemble average: Ensemble Count $= \frac{\text{Count}_{S_1} + \text{Count}_{S_2}}{2}$.

4.3 Experiments

4.3.1 Baselines and Datasets

To demonstrate the effectiveness and generalization of our approach, we integrate it into few recent DETR-based baselines: CountGD[8], CountSE[6] and CoLSR[1]. Our evaluation focuses on the performance gain (Δ) over the base models rather than standalone state-of-the-art results, demonstrating that CoLC serves as a consistent performance multiplier. We test this across datasets exhibiting structural arrangements: CARPK [58], PUCPR+ [59, 58], SKU110K [60], and SurgCount-HD [1].

CARPK [58] is a drone-based car parking dataset containing 989 training and 459 test images with bounding box annotations. We use "car" as the text prompt for training our models with CoLA.

PUCPR+ [59, 58] dataset (Pontifical Catholic University of Parana+) contains 10th-floor views of parking lots with bounding box annotations. It comprises 95 training images and 25 test images. We use "car" as the text prompt for training with CoLA.

SKU-110K [60] consists of images of supermarket store shelves with dense bounding box annotations. It contains 8,233 training and 2,941 test images. We use "retail item" as the text prompt for this dataset.

SurgCount-HD [1] is a surgical instrument dataset captured in high-density operating room settings with bounding box annotations of instrument handles. It includes 1,236 training and 228 test images. We use "circular surgical instrument" as the text prompt.

4.3.2 Implementation Details

Training. During training, the base model is frozen and only the CoLC parameters are optimized using the Adam optimizer. The position embeddings and multi-head projection layers are trained with a learning rate of 1×10^{-4}, while the scalar chain gain γ_{chain} uses 1×10^{-3}. We train for 30 epochs: 10-epochs of linear warmup followed by a 20-epoch Cosine Annealing schedule. Across all datasets and baselines, γ_{chain} is initialized to 1.0 and the position range L_{chain} to 16. All other hyperparameters remain identical to their respective baseline configurations.

Additionally, our framework relies on two key threshold parameters: δ_c, which filters active chain candidates from the initial query proposals, and σ_f, which serves as the final prediction threshold after the decoder. We initialize δ_c based on the original final threshold of the respective base model. We adopt this strategy based on the assumption that base vision models already generate high-quality object proposals but struggle to effectively utilize them for structured counting. The threshold configurations for each baseline and dataset are detailed in Table 4.1.

Table 4.1: Threshold settings (δ_c and σ_f) across different baseline models and datasets.

Baseline Model	**Dataset**	δ_c	σ_f
CountGD [8]	All Datasets	0.23	0.23
CountSE [6]	Default (All others)	0.35	0.35
CountSE [6]	CARPK	0.35	0.23
CoLSR [1]	All Datasets	0.23	0.26

Inference. At inference, the image and corresponding text prompt are processed to produce bi-directional counts after thresholding with σ_f. Since ground truth is not present to report $MAE_{GT-Select}$ (Sec. 4.3.2), we use a consensus-based post-processing operator $\mathcal{P}$ to consolidate the dual predictions and remove duplicates via point-based non-maxima suppression. Specifically, we aggregate our final predictions using a three-step postprocessing pipeline applied to the outputs of directions S_1 and S_2:

1. **Distance Computation:** We compute a pairwise Euclidean distance matrix $\mathbf{D} \in \mathbb{R}^{N_{S_1} \times M_{S_2}}$ between the two predicted point sets.

2. **Point Consensus:** Predictions from S_1 and S_2 are cross-referenced. If a point's nearest neighbor in the other set falls within a predefined distance threshold, the detections are merged.

3. **Duplicate Removal:** We apply Point Non-Maximum Suppression (NMS) to the merged set to filter out redundant points and establish the final count.

Evaluation Metrics. We use the standard Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) as evaluation metrics.

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|N_P - N_G|, \quad \text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(N_P - N_G)^2}, \tag{4.4}$$

where N is the number of image samples, N_P is the predicted count and N_G is the ground truth count for image N_i.

We also report GT-Select, a metric representing the model's performance when the optimal direction is selected for each individual image. This serves as an lower bound on error for our method and is formulated as:

$$\text{MAE}_{\textit{GT-Select}} = \frac{1}{N}\sum_{i=1}^{N}\min(|y_i - \hat{y}_{i,S_1}|, |y_i - \hat{y}_{i,S_2}|), \tag{4.5}$$

where y_i is the ground truth count and $\hat{y}_{i,S_1}, \hat{y}_{i,S_2}$ are the predicted counts for the two directions.

4.3.3 Quantitative Results

Table 4.2 illustrates the impact of CoLC when integrated into existing baselines. Across all four benchmarks, CoLC-enhanced models achieve comparable or superior MAE, demonstrating the module's ability to refine counting accuracy without degrading baseline performance.

4.3.4 Qualitative Results

Figures 4.3 and 4.4 provide qualitative comparisons. In dense or heavily occluded scenes, baseline models often suffer from local feature ambiguity and poor spatial reasoning, resulting in missed detections and double-counting. By enforcing a topological "chain-of-look", CoLC enforces stronger structural context to overcome these local ambiguities, improving overall detection stability.

Table 4.2: **Quantitative Evaluation on Structural Datasets.** We compare the vanilla baselines against our CoLC module across four counting strategies: starting from the closest point (S_1), the farthest point (S_2), the image-level Ensemble, and the theoretical optimal direction (GT-Select).

Method	CARPK		PUCPR+		SKU110K		SurgCount-HD	
	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓
Baseline 1: CountGD								
CountGD [8] (Zeroshot)	3.72	5.26	23.96	41.25	69.54	91.40	23.34	30.39
CountGD + CoLC$_{S_1}$	3.04	4.54	23.44	40.80	43.94	56.81	12.77	18.19
CountGD + CoLC$_{S_2}$	3.10	4.60	23.56	40.57	44.34	57.32	12.48	17.72
CountGD + CoLC$_{Ensemble}$	3.06	4.76	23.50	**40.15**	44.09	57.00	12.61	17.93
CountGD + CoLC$_{\text{GT-Select}}$	**2.94**	**4.45**	**23.24**	40.50	**42.30**	**55.53**	**12.04**	**17.45**
Baseline 2: CountSE								
CountSE [6] (Zeroshot)	2.79	4.2	8.00	20.81	27.12	46.62	19.01	25.25
CountSE + CoLC$_{S_1}$	2.59	4.07	7.88	20.77	24.22	41.15	7.82	10.18
CountSE + CoLC$_{S_2}$	2.56	4.02	7.92	20.76	24.38	41.46	7.72	10.30
CountSE + CoLC$_{Ensemble}$	2.57	4.03	7.90	20.76	24.29	41.29	7.72	10.18
CountSE + CoLC$_{\text{GT-Select}}$	**2.47**	**3.96**	**7.8**	**20.76**	**23.56**	**40.73**	**7.05**	**9.60**
Baseline 3: CoLSR								
CoLSR [1] (Zeroshot)	-	-	-	-	-	-	0.88	**1.27**
CoLSR + CoLC$_{S_1}$	-	-	-	-	-	-	0.9	1.33
CoLSR + CoLC$_{S_2}$	-	-	-	-	-	-	0.89	1.30
CoLSR + CoLC$_{Ensemble}$	-	-	-	-	-	-	0.89	1.31
CoLSR + CoLC$_{\text{GT-Select}}$	-	-	-	-	-	-	**0.87**	1.29

Table 4.3: **Error Analysis: Average Overcounts and Undercounts.** The average number of extra objects (Overcount) and missed objects (Undercount) per image across all datasets.

Method	CARPK		PUCPR+		SKU110K		SurgCount-HD	
	Over. ↓	Under. ↓	Over. ↓	Under. ↓	Over. ↓	Under. ↓	Over. ↓	Under. ↓
CountGD (Zeroshot)	**0.12**	3.59	23.68	0.28	**3.26**	66.27	23.08	**0.26**
CountGD + CoLC$_{S_1}$	0.32	**2.71**	**23.2**	**0.24**	25.59	**18.34**	**11.85**	0.91
CountSE (Zeroshot)	**0.31**	2.46	**6.12**	1.88	20.39	**6.74**	17.87	**1.14**
CountSE + CoLC$_{S_1}$	0.49	**2.09**	6.52	**1.36**	**15.12**	9.09	**3.72**	4.11
CoLSR (Zeroshot)	-	-	-	-	-	-	0.53	**0.35**
CoLSR + CoLC$_{S_1}$	-	-	-	-	-	-	**0.30**	0.60

Bi-Directional Counting. Figure 4.5 illustrates the impact of our bi-directional counting strategy. Selecting a different starting node produces a distinct greedy chain, thereby providing a complementary topological view that smooths out local heuristic errors.

4.3.5 Error Analysis

Table 4.3 decomposes the overall error into average absolute overcount and undercount per image, defined as:

$$\text{Overcount} = \frac{1}{N}\sum_{i=1}^{N}\max(0, \hat{y}_i - y_i), \qquad \text{Undercount} = \frac{1}{N}\sum_{i=1}^{N}\max(0, y_i - \hat{y}_i). \tag{4.6}$$

The metrics show that CoLC acts as a powerful adaptive regularizer, specifically targeting the dominant failure mode of the zero-shot baselines. In scenarios where the baseline suffers from severe missed detections, such as CountGD on the highly dense SKU110K dataset, CoLC drastically reduces undercounting from 66.27 to 18.34. In contrast, when the baseline overcounts, as seen in SurgCount-HD where it exceeds 23, CoLC effectively suppresses these false positives and reduces them by up to 65% for CountSE. Even though fixing the model's main weakness sometimes causes a minor increase in the opposite error type, the total error drops significantly, resulting in a much more accurate final count.

4.3.6 Mechanistic Analysis

Impact of RPE Bias on Attention Weights. To quantify the extent of CoLC's impact on the base model, we measure the Mean Query Peak, defined as the average maximum shift in post-Softmax attention probability for active object queries. Essentially, it represents the percentage of the attention budget that the Counting reallocates per object to enforce spatial structure. Table 4.4 reports the Mean Query Peak across the six decoder self-attention layers. We observe two consistent behaviors: **(1)** Regardless of the dataset, the impact follows a

Table 4.4: **Mean Query Peak (%) across Decoder Self-Attention Layers.** Text colors indicate intervention intensity: Low, Medium, and High. The "⋀" pattern across each row illustrates how CoLC peaks in the middle layers before fading away. Analysis performed using $\texttt{CountSE+CoLC}_{S_1}$.

Dataset	Layer 1	Layer 2	Layer 3	Layer 4	Layer 5	Layer 6
PUCPR+	0.77	**2.87**	1.31	1.69	0.83	0.43
SKU110K	3.08	7.18	6.25	**11.25**	2.99	2.11
CARPK	15.85	**29.94**	17.15	26.32	11.57	8.62
SurgCount-HD	7.61	18.00	14.32	**18.80**	6.86	4.30

Table 4.5: **Ablation on Position Embedding Chain Length.** Impact of the chain length (L_{chain}) on counting performance. **Bold** indicates the best performance, while underline indicates the worst. Lower-density datasets (CARPK, PUCPR+) exhibit a non-linear trend, degrading initially before recovering at the shortest chain length. Analysis performed using $\texttt{CountSE+CoLC}_{S_1}$.

	CARPK		PUCPR+		SKU110K		SurgCount-HD	
L_{chain}	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓
16	2.59	4.07	**7.88**	**20.77**	**24.22**	**41.15**	**10.05**	**13.17**
8	2.60	4.06	8.08	20.78	24.68	41.93	10.51	14.06
4	2.62	4.10	8.12	20.88	24.72	41.98	10.53	14.14
2	**2.57**	**4.04**	8.00	20.86	24.84	42.26	10.72	14.28

distinct curve. The Counting applies minimal shifts in Layer 1 and peaks in the middle layers (L2 and L4) to prioritize spatial grouping. The impact fades in the final layers, ensuring that the structural bias does not override the base model's ability to regress precise object coordinates.**(2)** In datasets with high visual uncertainty or low contrast (e.g., CARPK peaking at 29.94%), CoLC alters attention to override failing visual features. In contrast, in distinct, high-contrast environments (e.g., PUCPR+ peaking at 2.87%), the Counting applies minimal correction to the already confident base model.

4.3.7 RPE Bias Distribution Analysis

While the mechanistic analysis provides insight into how the RCPE module biases attention weights, it is also important to investigate exactly *which* queries receive this routed attention. We conduct an experiment to quantify the percentage of active chain queries whose maximum attention shift (induced by the RPE_{bias}) successfully targets another active chain candidate rather than background regions or self-attention. This hit-rate determines whether the model diverts its primary focus meaningfully toward structurally relevant objects. Table 4.6 illustrates these attention routing percentages, confirming that the RPE_{bias} acts as a highly effective structural prior.

Table 4.6: The percentage of active queries whose maximum attention shift (peak shift) successfully targets another valid chain candidate during decoder self-attention. Results evaluate the CountSE baseline across multiple datasets.

Dataset Source	**Layer 1**	**Layer 2**	**Layer 3**	**Layer 4**	**Layer 5**	**Layer 6**
SKU110K	74.55	38.74	51.74	45.76	72.17	89.22
PUCPR+	65.12	45.00	62.26	42.10	60.57	80.29
CARPK	58.21	45.48	38.33	32.78	65.42	68.01
SurgCount-HD	76.96	60.90	56.12	55.60	63.43	80.33

4.3.8 Ablation Study

We ablate the visual chain length ($L_{chain} \in \{2, 4, 8, 16\}$) using our best-performing CountSE + CoLC_{S_1} setting to evaluate its impact on spatial reasoning (Table 4.5). We observe that the optimal chain length strongly correlates with the density and structural complexity of the dataset. On high-density datasets like SKU110K and SurgCount-HD, performance degrades monotonically as L_{chain} decreases. Objects in SurgCount-HD, for example, suffer from extreme occlusion, specularity and

dense packing along structural lines, resolving these visual ambiguities requires global sequence modeling ($L_{chain} = 16$). Similarly, the densely clustered retail items in SKU110K benefit significantly from the broader spatial context provided by the longest chain. In contrast, CARPK has relatively isolated objects in predictable grids, where simple local spatial smoothing ($L_{chain} = 2$) performs best. Given the marginal improvement in shorter chains, we set $L_{chain} = 16$ as the default to ensure robustness in challenging, highly clustered environments.

4.4 Limitations

The current Chain Constructor (CC) is largely heuristic-driven, making the resulting topological order sensitive to the choice of the initial node. An inappropriate or noisy starting point can produce disordered chains that connect distant objects or form "zig-zag" patterns, disrupting the spatial and semantic continuity necessary for accurate counting. Moreover, this work focuses on closed-set structured object counting and does not address the more challenging open-set setting, where object categories may vary beyond predefined classes.

4.5 Chapter Summary

In this chapter, we introduced Chain-of-Look Counting (CoLC), a plug-and-play framework for structured object counting that explicitly models spatial layout through sequential aggregation. Without modifying the base counter, CoLC consistently improves accuracy across diverse structured object counting benchmarks. Its bi-directional traversal enhances robustness and explicit intermediate steps provide transparent counting trajectories. Overall, our results demonstrate

that incorporating spatial structure via sequence-aware reasoning is a simple yet effective strategy for reliable and transparent structured counting. Looking ahead, we will develop a learning-based chain constructor to adaptively handle complex layouts and high-density occlusions. We also aim to extend CoLC to more challenging open-set structured counting scenarios.

Figure 4.3: **Qualitative comparison on CARPK, PUCPR+ & SurgCount-HD datasets.** Each row displays the Ground Truth annotations (left), the Baseline predictions (middle), and our CoLC predictions (right). Regions where a model fails are highlighted with a red indicator ( ) and corresponding successful with a green indicator ( ).

Figure 4.4: **Qualitative comparison on SKU110K dataset.** Each row displays the Ground Truth annotations (left), the Baseline predictions (middle), and our CoLC predictions (right). Regions where a model fails are highlighted with a red indicator ( ) and corresponding successful with a green indicator ( ).

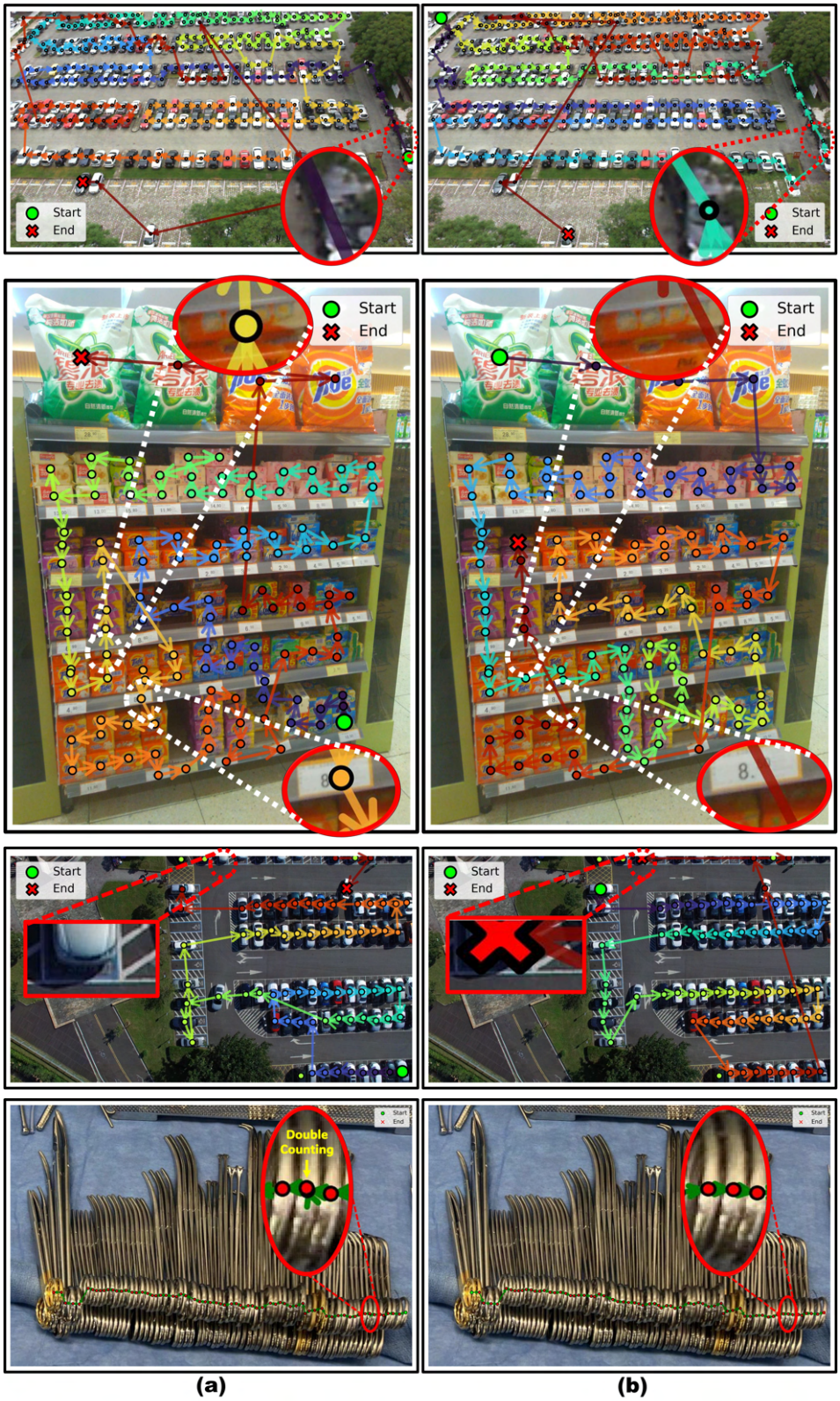

Figure 4.5: **Qualitative of Bi-directional Counting.** Each row displays the S2 Direction CoLC_{S_2} (left), and the S1 Direction CoLC_{S_1} (right) results.

Chapter 5

Future Work

Counting is one of the fundamental tasks in computer vision. As artificial intelligence advances, the native ability to count objects in images and videos has become increasingly desirable. To count at scale, models need to align the counting process with human cognition, as demonstrated in this thesis. However, there are still many challenges that need to be addressed in order to achieve this.

The Chain-of-Look (CoL) counting framework works under the assumption that the underlying encoder is capable of detecting objects accurately. This assumption fails in zero-shot scenarios where the model has not seen the objects during training. Architectural changes, such as unfreezing the encoder and carefully designing it to induce the chain bias, can help improve detection performance. Furthermore, the current CoL framework relies on domain-specific objectives or heuristic-based chain constructors to generate the topological order. Future research should focus on the development of a learning-based constructor to adaptively navigate complex scenes and handle high-density occlusions more robustly, while also exploring the integration of CoL with advanced reasoning capabilities (Large Language Models (LLMs)) to enable more sophisticated

counting strategies.

Another key direction CoL touches upon is the verification and interpretability of the counting process, an aspect often overlooked in object counting research. Building such inherently transparent models is key to deploying production-ready systems in safety-critical applications such as autonomous driving and medical imaging. 3D Digital Twin-based verification is one such direction that can be explored. We can also take this a step further by introducing physical constraints. By teaching the model basic physics rules, such as the fact that two solid objects cannot occupy the same space, the system can learn to automatically filter out impossible predictions. Additionally, integrating CoL with LLMs can help generate natural language explanations for the counting process, making it more transparent and interpretable. Finally, coupling uncertainty quantification methods with the counting process can help identify and prevent risks and failures associated with deploying such counting models.

Chapter 6

Conclusion

This thesis has explored the critical role of spatial reasoning in object counting. By taking inspiration from human cognition, specifically how humans sequentially traverse a scene, forming a mental chain to group and count objects in order to avoid omissions and double-counting, this research has shown that inducing structural biases into computer vision models significantly enhances counting accuracy, interpretability, and verifiability.

First, a domain-specific approach was taken by developing the Chain-of-Look Spatial Reasoning (CoLSR) framework for counting dense surgical instruments in operating rooms. By enforcing a visual chain through a novel neighboring loss and a class-specific learnable (CSL) token-based contrastive feature enhancer, the model effectively overcame the challenges of high visual similarity and severe occlusion, outperforming existing state-of-the-art methods. Furthermore, this research introduced the SurgCount-HD dataset, providing a rigorous benchmark for future clinical counting research.

Building upon these observations, the subsequent work generalized the approach to diverse, structured scenes with the Chain-of-Look Counting (CoLC)

framework. CoLC, a lightweight and versatile plug-and-play module, demonstrated that current DETR-based counting models possess an inherent affinity for sequential reasoning, supporting the broader usefulness of the chain-of-look paradigm. The bi-directional sequential aggregation mechanism consistently improves the performance of base counters without requiring architectural modifications, while also providing a verifiable and interpretable counting process.

In conclusion, the methodologies presented in this thesis demonstrate that counting at scale requires not just strong perceptual detectors, but also a principled mechanism to organize detections into a coherent, spatially-aware topological structure. By aligning machine counting with human visual reasoning, this thesis lays a robust foundation for building reliable, transparent, and structurally aware counting systems.

Bibliography

[1] Rishikesh Bhyri, Brian R Quaranto, Junsong Yuan, Peter CW Kim, and Nan Xi. Chain-of-look spatial reasoning for dense surgical instrument counting. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8521–8530, 2026. © 2026 IEEE. Reprinted, with permission, from *ishikesh Bhyri, Brian R Quaranto, Junsong Yuan, Peter CW Kim, and Nan Xi. Chain-of-look spatial reasoning for dense surgical instrument counting. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 8521-8530, 2026.*

[2] Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching clip to count to ten. *arXiv preprint arXiv:2302.12066*, 2023.

[3] Google. Gemini 3. https://aistudio.google.com/models/gemini-3, 2026.

[4] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pages 38–55. Springer, 2024.

[5] Siyang Dai, Jun Liu, and Ngai-Man Cheung. Referring expression counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16985–16995, June 2024.

[6] Shuai Liu, Peng Zhang, Shiwei Zhang, and Wei Ke. Countse: Soft exemplar open-set object counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21536–21546, 2025.

[7] Jer Pelhan, Alan Lukežic, Vitjan Zavrtanik, and Matej Kristan. Dave - a detect-and-verify paradigm for low-shot counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23293–23302, June 2024.

[8] Niki Amini-Naieni, Tengda Han, and Andrew Zisserman. Countgd: Multi-modal open-world counting. *Advances in Neural Information Processing Systems*, 37:48810–48837, 2024.

[9] Anindya Mondal, Sauradip Nag, Xiatian Zhu, and Anjan Dutta. Omnicount: Multi-label object counting with semantic-geometric priors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 19537–19545, 2025.

[10] Ziqiang Shi, Rujie Liu, Jun Takahashi, and Shan Jiang. Truecount: Improving open-world object counting with visual-language models and dynamic multi-modal inputs. In *Proceedings of the 33rd ACM International Conference on Multimedia*, MM '25, page 1764–1773, New York, NY, USA, 2025. Association for Computing Machinery.

[11] Mengmi Zhang Zenglin Shi, Ying Sun. Training-free object counting with prompts. In *WACV*, 2024.

[12] Nikola Djukic, Alan Lukežič, Vitjan Zavrtanik, and Matej Kristan. A low-shot object counting network with iterative prototype adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 18872–18881, October 2023.

[13] Zhuoxuan Peng and S.-H. Gary Chan. Single domain generalization for crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2024)*, 2024.

[14] Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. In *Proceedings of the 24th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'10, page 1324–1332, Red Hook, NY, USA, 2010. Curran Associates Inc.

[15] Zhi-Qi Cheng, Qi Dai, Hong Li, Jingkuan Song, Xiao Wu, and Alexander G. Hauptmann. Rethinking spatial invariance of convolutional networks for object counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19638–19648, June 2022.

[16] Mingyue Guo, Li Yuan, Zhaoyi Yan, Binghui Chen, Yaowei Wang, and Qixiang Ye. Regressor-segmenter mutual prompt learning for crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28380–28389, 2024.

[17] Xiaofei Hui, Qian Wu, Hossein Rahmani, and Jun Liu. Class-agnostic object counting with text-to-image diffusion model. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors,

Computer Vision – ECCV 2024, pages 1–18, Cham, 2025. Springer Nature Switzerland.

[18] Yanda Meng, Hongrun Zhang, Yitian Zhao, Xiaoyun Yang, Xuesheng Qian, Xiaowei Huang, and Yalin Zheng. Spatial uncertainty-aware semi-supervised crowd counting. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15549–15559, 2021.

[19] Yasiru Ranasinghe, Nithin Gopalakrishnan Nair, Wele Gedara Chaminda Bandara, and Vishal M Patel. Crowddiff: Multi-hypothesis crowd density estimation using diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12809–12819, 2024.

[20] Viresh Ranjan, Udbhav Sharma, Thu Nguyen, and Minh Hoai. Learning to count everything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[21] Giacomo Pacini, Lorenzo Bianchi, Luca Ciampi, Nicola Messina, Giuseppe Amato, and Fabrizio Falchi. Countingdino: A training-free pipeline for class-agnostic counting using unsupervised backbones. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 806–815, March 2026.

[22] Muhammad Ibraheem Siddiqui and Muhammad Haris Khan. Countzes: Counting via zero-shot exemplar selection. *arXiv preprint arXiv:2512.16415*, 2025.

[23] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[24] Rochel Gelman, Charles R Gallistel, and Rochel Gelman. *The child's understanding of number*. Harvard University Press, 2009.

[25] Gordon Logan, Jane Zbrodoff, and Xingshan Li. Do the eyes count? the role of eye movements in visual enumeration. *Journal of Vision*, 8(6):115–115, 2008.

[26] Joseph Paul Cohen, Genevieve Boucher, Craig A. Glastonbury, Henry Z. Lo, and Yoshua Bengio. Count-ception: Counting by fully convolutional redundant counting. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017.

[27] Yuhong Li, Xiaofan Zhang, and Deming Chen. 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.

[28] Khac-Hoai Nam Bui, Hongsuk Yi, and Jiho Cho. A vehicle counts by class framework using distinguished regions tracking at multiple intersections. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2466–2474, 2020.

[29] Maryam Rahnemoonfar and Clay Sheppard. Deep count: Fruit counting based on deep simulated learning. *Sensors*, 17(4), 2017.

[30] Andreas Michel, Wolfgang Gross, Fabian Schenkel, and Wolfgang Middelmann. Class-aware object counting. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 469–478, 2022.

[31] Weidi Xie, J. Alison Noble, and Andrew Zisserman. Microscopy cell counting and detection with fully convolutional regression networks. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 6(3):283–292, 2018.

[32] Liu Chang, Zhong Yujie, Zisserman Andrew, and Xie Weidi. Countr: Transformer-based generalised visual counting. In *British Machine Vision Conference (BMVC)*, 2022.

[33] Shiwei Zhang, Qi Zhou, and Wei Ke. Enhancing zero-shot object counting via text-guided local ranking and number-evoked global attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 21097–21106, October 2025.

[34] Dongze Lian, Jing Li, Jia Zheng, Weixin Luo, and Shenghua Gao. Density map regression guided detection network for rgb-d crowd counting and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1821–1830, 2019.

[35] Dongze Lian, Xianing Chen, Jing Li, Weixin Luo, and Shenghua Gao. Locating and counting heads in crowds with a depth prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9056–9072, 2021.

[36] Corentin Dumery, Noa Etté, Aoxiang Fan, Ren Li, Jingyi Xu, Hieu Le, and Pascal Fua. Counting Stacked Objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[37] Huang Zhizhong, Dai Mingliang, Zhang Yi, Zhang Junping, and Shan Hongming. Point, segment and count: A generalized framework for object counting. In *CVPR*, 2024.

[38] Zhiyuan You, Kai Yang, Wenhan Luo, Xin Lu, Lei Cui, and Xinyi Le. Few-shot object counting with similarity-aware feature enhancement. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6315–6324, 2023.

[39] Yifeng Huang, Duc Duy Nguyen, Lam Nguyen, Cuong Pham, and Minh Hoai. Count what you want: exemplar identification and few-shot counting of human actions in the wild. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 10057–10065, 2024.

[40] Min Shi, Hao Lu, Chen Feng, Chengxin Liu, and Zhiguo Cao. Represent, compare, and learn: A similarity-aware framework for class-agnostic counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9529–9538, 2022.

[41] Thanh Nguyen, Chau Pham, Khoi Nguyen, and Minh Hoai. Few-shot object counting and detection. In *European Conference on Computer Vision*, pages 348–365. Springer, 2022.

[42] Shuo-Diao Yang, Hung-Ting Su, Winston H Hsu, and Wen-Chin Chen. Class-agnostic few-shot object counting. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 870–878, 2021.

[43] Nan Xi, Jingjing Meng, and Junsong Yuan. Chain-of-look prompting for verb-centric surgical triplet recognition in endoscopic videos. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 5007–5016, 2023.

[44] Nan Xi, Jingjing Meng, and Junsong Yuan. Open set video hoi detection from action-centric chain-of-look prompting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3079–3089, 2023.

[45] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning, 2022.

[46] Hantao Yao, Rui Zhang, Lu Yu, Yongdong Zhang, and Changsheng Xu. Sep: Self-enhanced prompt tuning for visual-language model, 2024.

[47] Seunggu Kang, WonJun Moon, Euiyeon Kim, and Jae-Pil Heo. Vlcounter: Text-aware visual representation for zero-shot object counting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2714–2722, 2024.

[48] Roboflow Team. Roboflow: Computer vision platform. https://roboflow.com, 2025.

[49] OpenAI. Gpt5. https://openai.com/gpt-5/, 2025.

[50] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[51] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[52] Ricardo Guerrero-Gómez-Olmedo, Beatriz Torre-Jiménez, Roberto López-Sastre, Saturnino Maldonado-Bascón, and Daniel Onoro-Rubio. Extremely overlapping vehicle counting. In *Iberian conference on pattern recognition and image analysis*, pages 423–431. Springer, 2015.

[53] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[54] Yi-Xin Huang, Hou-I Liu, Hong-Han Shuai, and Wen-Huang Cheng. Dq-detr: Detr with dynamic query for tiny object detection, 2024.

[55] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

[56] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[57] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 464–468, 2018.

[58] Meng-Ru Hsieh, Yen-Liang Lin, and Winston H. Hsu. Drone-based object counting by spatially regularized regional proposal networks. In *The IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017.

[59] Paulo RL De Almeida, Luiz S Oliveira, Alceu S Britto Jr, Eunelson J Silva Jr, and Alessandro L Koerich. Pklot–a robust dataset for parking lot classification. *Expert Systems with Applications*, 42(11):4937–4949, 2015.

[60] Eran Goldman, Roei Herzig, Aviv Eisenschtat, Jacob Goldberger, and Tal Hassner. Precise detection in densely packed scenes. In *Proc. Conf. Comput. Vision Pattern Recognition (CVPR)*, 2019.