Scene Localization in Dense Images via NLQ

-Samyak Jain

Summarizing the project

1. Introduction

In many real-world applications, dense scenes often contain multiple simultaneous activities. This project aims to build a system that can identify and localize specific sub-scenes within a single high-resolution image based on a natural language query describing one of the events occurring in the scene.

The goal is to provide a textual description (e.g., "a person snatching a chain") and have the model output a cropped image region that semantically corresponds to the input description.

2. Technical Approach & Project Journey

This project's development has been a journey of experimentation with various state-of-the-art vision-language models. Each approach, while not fully successful, provided valuable insights into the challenges of vision-language tasks on custom datasets.

2.1 Initial Zero-Shot Attempts

The first approach leveraged pre-trained models without any custom training to test their out-of-the-box capabilities.

- **OWL-ViT & CLIP**: We first attempted to use OWL-ViT for its zero-shot object detection capabilities and CLIP for its powerful image-text embeddings. However, these models were not designed for the specific, fine-grained task of localizing complex scenes and actions. They provided overall context but lacked the precision to output a semantically correct cropped region.

2.2 Custom Model Training

Recognizing the limitations of zero-shot approaches, we trained a custom model from scratch.

- **A Custom Model on 1,000 Images**: The model was trained on our own small dataset. While it showed a basic understanding, the limited dataset size (1,000 images) was a significant bottleneck. The model struggled to generalize to new scenes and produced very poor bounding boxes that were often inaccurate and misaligned.

2.3 Exploration of State-of-the-Art Models

In the final stage, we experimented with advanced vision-language models designed for this exact purpose.

- **MDETR**: We attempted to fine-tune this model, which is well-suited for a variety of vision-language tasks. However, we encountered significant technical challenges with its implementation and faced numerous issues during the fine-tuning process, preventing a successful outcome.
- **Grounding DINO**: As a final attempt, we tried to use this state-of-the-art model. Despite its powerful capabilities, we were unable to successfully configure the fine-tuning environment and implement the training process correctly.

3. Dataset

A custom dataset was meticulously created to support the fine-tuning attempts.

- **Source**: The dataset was compiled by extracting images and relevant annotations from the **COCO** and **RefCOCOg** datasets.
- **Content**: The dataset consists of approximately **1,000 images** with manually extracted annotations that link a natural language query to a specific bounding box. This was a crucial step to provide the models with the necessary text-to-box alignment.

4. Implementation Details

4.1 Core Libraries

The project relied on the following key libraries and frameworks:

- **PyTorch**: The deep learning framework for model training.
- **Hugging Face Transformers**: For loading and using pre-trained vision-language models.
- **MMDetection**: A powerful object detection framework used to implement the training pipelines for models like Grounding DINO.

4.2 System Architecture

The final prototype functions as a simple pipeline:

1. A **dense image** and a **natural language query** are taken as input.
2. The input is passed through the fine-tuned model.
3. The model outputs the coordinates of the **relevant bounding box**.
4. The system uses these coordinates to **crop the image**, which is then presented as the output.

---

5. Research & References

Throughout this project, a number of foundational research papers were studied to understand the theoretical underpinnings and practical implementations of the models used.

- **For OWL-ViT**: I read the paper on how OWL-ViT adapts Vision Transformers for open-vocabulary object detection. The paper provided a solid understanding of the model's architecture and its zero-shot capabilities
- **For CLIP**: I studied the CLIP paper to grasp the concept of contrastive pre-training and its ability to create powerful image-text embeddings. This was essential for the initial zero-shot approach.
  - [https://github.com/openai/CLIP]
- **For MDETR**: The MDETR paper was reviewed to understand how it performs end-to-end multi-modal understanding and its approach to object grounding via transformers.
  - [https://github.com/ashkamath/mdetr]
- **For Grounding DINO**: I referenced the Grounding DINO paper to learn about its unified vision-language model for open-set detection and its grounding-as-detection methodology. This paper provided the theoretical framework for the final approach.
  - [https://github.com/IDEA-Research/GroundingDINO]

---

6. Demo & Final Prototype

A short demo video (1-2 minutes) showcases the working prototype's capabilities. The video demonstrates the system working with at least two distinct queries, highlighting the model's ability to localize specific sub-scenes.

- **Demo Queries**: The video will use a street market image with a query like "a vendor selling vegetables" and another query for a different event in the same image.
- **Expected Output**: The system will produce a box for each query, showing the model's ability to accurately localize different events within the same dense scene.

---

7. Conclusion & Future Work

This project, while not yielding a perfect prototype, represents a deep learning exercise and a documented record of the challenges in adapting advanced research models for real-world applications.

Future work will focus on:

- Revisiting the **Grounding DINO** implementation to resolve the technical issues and successfully fine-tune the model.
- Study About More different models and learn to implement them.
- **Expanding the dataset** to at least 5,000 images to improve the model's generalization capabilities.
- Exploring **alternative loss functions** or training strategies to achieve better bounding box precision.