Sketch Query Guided Object Detection

Deep Wilson Aricatt

Thesis Report for Master of Science in Computer Vision, Robotics and Machine Learning

from the

University of Surrey

Department of Electrical and Electronic Engineering

Faculty of Engineering and Physical Sciences

University of Surrey

Guildford, Surrey, GU2 7XH, UK

September 2023

Supervised by: Yi-Zhe Song

DECLARATION OF ORIGINALITY

I confirm that the project dissertation I am submitting is entirely my own work and that any material used from other sources has been clearly identified and properly acknowledged and referenced. In submitting this final version of my report to the JISC anti-plagiarism software resource, I confirm that my work does not contravene the university regulations on plagiarism as described in the Student Handbook. In so doing I also acknowledge that I may be held to account for any particular instances of uncited work detected by the JISC anti-plagiarism software, or as may be found by the project examiner or project organiser. I also understand that if an allegation of plagiarism is upheld via an Academic Misconduct Hearing, then I may forfeit any credit for this module or a more severe penalty may be agreed.

MSc Dissertation Title

Author name: Deep Wilson Aricatt

Author Signature: Date: 01 September 2023

Supervisor's name: Yi-Zhe Song

Co-supervisor's name: Pinaki Nath Chowdhury

ACKNOWLEDGEMENTS

Working on this thesis has been an incredibly fulfilling experience. It provided me with the opportunity to bridge the gap between theoretical concepts and real-world applications, allowing me to witness the impact of multimodal applications in practical scenarios. Throughout this journey, I encountered many challenges that pushed the boundaries of my understanding and problem-solving skills.

I would like to express my deepest gratitude to my supervisor, Prof. Yi-Zhe Song, for his guidance, and invaluable mentorship throughout the course of my first semester and thesis. His lectures on Advanced Computer Vision strengthened my concepts and made me contemplate deeper and more complex topics within computer vision. As he mentioned, in today's rapidly evolving research landscape, it is imperative to stick to fundamentals and this philosophy has significantly inspired me to utilize approaches like DETR, which represents a fundamental shift from geometric-based object detection techniques.

Additionally, I would like to thank Pinaki Nath Chowdhury for co-supervising my thesis. His extensive research experience undoubtedly enriched my work from day one, as he generously shared his ideas and research insights.

Furthermore, I extend my sincere appreciation to the folks at the Centre for Vision, Speech and Signal Processing of the University of Surrey (CVSSP) for their valuable inputs and insightful discussions. I would also like to thank the University of Surrey for providing the necessary GPU resources to carry out my experiments.

Finally, I would like to thank my family for their love, support and constant encouragement. I would like to dedicate this thesis to my nephew Ethan who has brought me so much joy ever since we welcomed him into our lives this year.

WORD COUNT

Number of Pages: 55

Number of Words: 10037

ABSTRACT

Sketches have been widely utilized throughout history for various purposes, from conveying ideas to recording observations. Despite the rise of language and digital media, the unparalleled expressiveness of sketches remains evident, often prompting us to resort to pen and paper or digital tools to visually capture our thoughts to convey minute details not possible through other modalities like text and speech.

Over the past decade, sketch research has flourished, covering an array of tasks ranging from traditional classification and synthesis to more specialized areas such as visual abstraction modeling [87], style transfer [106], and continuous stroke fitting [26]. One area that has garnered substantial attention is sketch-based image retrieval (SBIR), where sketches are used to retrieve relevant images from databases (content-based image retrieval). Fine-grained sketch-based image retrieval (FG-SBIR) has notably emerged as a significant focus, emphasizing nuanced details within sketches.

However, amidst the strides made in the field, we recognize a particular gap in the research landscape. While sketches' potential for image retrieval has been extensively explored, their applicability in object detection tasks has received comparatively less attention. Recent research [130], [24] and [102] has naturally evolved from SBIR to the more challenging task of sketch-guided object localization (SGOL) [102], [131]. SGOL entails precisely identifying and localizing objects within images based on sketches, a task that holds immense potential in enhancing various applications.

However, all the prior works focus on object detection using a single sketch patch at inference time. This limitation restricts end-users to detecting only simple instances. For example if a user draws a sketch of a zebra, the algorithm will localize all instances of zebras within the photo. This brings attention to a crucial issue: the inability to detect objects within natural images with any form of spatial awareness. For instance, users might be interested in detecting complex scenes [24] such as a "dog" to the right of a "person" or a "group of 3 zebras together", involving multiple objects with meaningful spatial alignment.

In this study, for the first time, we address this limitation by introducing a modified version of DETR [16]. Our approach incorporates a query canvas that empowers end users to draw multiple sketch instances. This allows for the detection of objects while taking their spatial alignment into account.

Moreover, our ablation studies reveal that the decoder conditioning, as proposed by [102] is unnecessary for localizing instances. We also establish that simple cross-modality fusion techniques, such as addition, suffice to attain sufficiently accurate results as opposed to more complicated methods.

Additionally, we introduce a cross-modality encoder block as a viable alternative to both addition and concatenation. This approach aims to enhance the accuracy of detecting multiple sketch instances without requiring any decoder conditioning.

Our main contributions include:

- **Detection with Spatial Awareness:** Our approach incorporates a query canvas that empowers end users to draw multiple sketch instances. This allows for the detection of objects while taking their spatial alignment into account.
- **Alignment of Photos and Sketches:**
  - Utilizing Concatenation or Addition Operator: We achieve photo-sketch alignment through fusion techniques such as concatenation or the addition operator.
  - Cross-attention Encoder block: We introduce a cross-modality encoder block as a promising alternative to both addition and concatenation, yielding improved accuracy.
- **Extending to Unseen Classes:** We use a simple trick of redefining the labels as either "object" or "no-object", allowing for a broader generalization across previously unseen categories.
- **Open source code base:** We open-source our code base so that fellow researchers can conduct further experiments with minimal downtime. The code base is a modification of the official DETR [16] implementation and is available at `https://github.com/deepwilson/detr`.

CONTENTS

LIST OF FIGURES

1 INTRODUCTION

The use of sketches as a means of communication dates back centuries and offers a unique and versatile way of conveying information. Sketches are expressive in nature, which enables them to encapsulate minute visual cues, while simultaneously maintaining a sparse structure. Other modalities may not be able to capture such intricate details which makes sketches a suitable modality for varied downstream tasks. With the emergence of touchscreen and digital pen devices, sketches have become pervasive for a broad range of tasks, including retrieval [30], object localization [130] and detection [24] in natural images.

1.1 Sketch-based Image Retrieval

Among various tasks centered around sketch modality, considerable attention has been directed towards sketch-based image retrieval (SBIR) with notable contributions from [30, 153, 5]. SBIR, a prominent research area within content-based image retrieval (CBIR) enables users to search for images using free-hand sketches. Typically, input sketches are characterized by a high level of abstraction and provide a rough representation of the overall shape and key local features of the target object or scene. In contrast, the gallery images in SBIR datasets often consist of realistic photographs or intricate artworks, making them significantly different from the input sketches. The primary goal of SBIR is to locate images that share similarities in both the overall shape and salient local details with the input sketch. SBIR frameworks signify a notable advancement in the practical application of sketches within real-world contexts. SBIR was initially framed as a category-level retrieval problem. However, it soon became evident that the primary advantage of sketches over text or tag-based retrieval lay in their ability to convey fine-grained details [6]. Consequently, this shifted the focus towards fine-grained SBIR, which aims to retrieve a specific photograph within a gallery placing a strong emphasis on capturing and retrieving highly detailed information within images. While traditional SBIR primarily targets category-level retrieval, where sketches are used to identify broad object or scene categories, fine-grained SBIR takes the process a step further retrieving gallery images by paying attention to minute details.

1.2 Sketch-based Object Detection

Recent works though limited in number have logically progressed beyond conventional SBIR and FG-SBIR, transitioning towards the detection/localization of objects within images, leveraging

sketches as guides [130, 102, 131]. A visual representation of this concept is depicted in Figure 1.1, where three distinct sketches are localized within a single natural image [102].

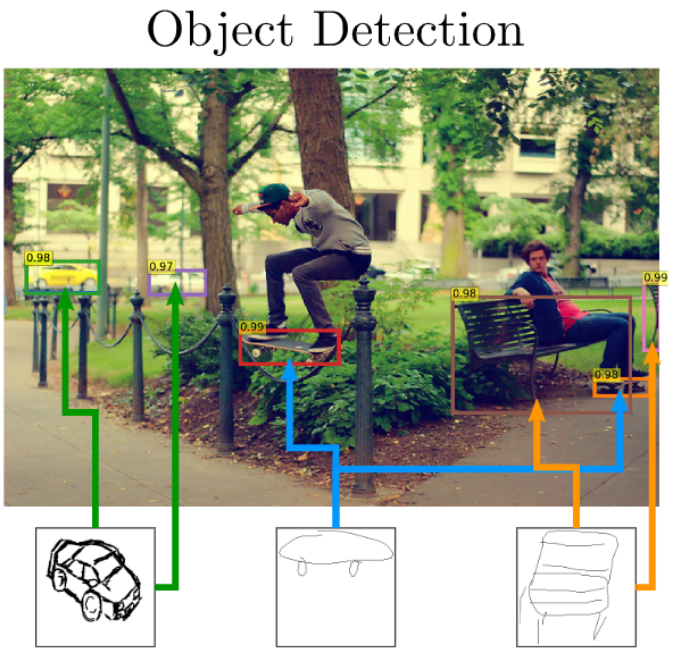

Figure 1.1: Example of object localization

Sketch-based object detection holds significant promise, particularly due to its potential real-world applications. While other modalities like text can be utilized, sketches offer several advantages, including the ability to convey intricate details that may be challenging to express through text descriptions.

Some of the advantages of using sketches over text for retrieval includes:

- **Visual Precision**: Sketches excel in providing precise and detailed representations of objects or scenes, especially in scenarios requiring fine-grained object localization.
- **Language Independence**: Sketches are universally understandable, transcending language barriers and making them accessible to a global audience. Text queries, in contrast, may encounter linguistic complexities or necessitate translations in multilingual settings.
- **Simplicity**: Sketches are often simpler to create than textual descriptions. Users may find it more straightforward to sketch a rough representation of an object rather than struggling to articulate it in words, particularly for intricate or abstract visual concepts.

- **Reduced Ambiguity**: Text-based queries can sometimes introduce ambiguity or require interpretation. Sketches mitigate this issue by offering a direct visual representation of the user's search intent.

1.3 Current approaches

1.3.1 Sketch-guided object localization in natural images

Tripathi et al. [130] first proposed a framework to tackle such a problem in terms of one-shot object detection given a sketch as a query. However, several limitations have been pointed out by [24] with respect to the problem definition as well as architectural designs. Firstly, instead of fine-grained matching, a sketch was used to specify the object category (which is easier via text/keyword [43, 86]), thus overlooking the potential of the sketch to model fine-grained details. Additionally, the detection pipeline in this work is based on traditional object detection pipelines such as Faster R-CNN. Such code-bases have a lot of hand-designed components like a non-maximum suppression procedure or anchor generation that explicitly encode prior knowledge, making it difficult to run.

We attempted to use the code-base provided by Tripathi et al. [130], however, we couldn't run the code due to conflicts in dependencies. Our investigation led us to consider the feasibility of adopting a straightforward yet effective framework, such as DETR (Data-efficient Image Transformer) [16], for the task of object detection.

1.3.2 Localizing Infinity-shaped fishes: Sketch-guided object localization in the wild

During our literature survey, we came across the work of Pau et al. [102]. They employed the existing DETR framework [16] that eliminated the need for hand-crafted components in object detection and detected objects from photos using sketch queries. A notable innovation in their approach named *Sketch-DETR* was the introduction of a technique termed *object query conditioning*. In this method, the features extracted from sketch queries were jointly fed into the transformer decoder alongside the query objects. The primary motivation behind this conditioning scheme was to provide the query object with not only spatial information but also contextual information regarding the content it should search within the image [102].

1.4 Shortcomings of Current Approaches and Solution Approach

1.4.1 Can process only a single query at a time

A significant limitation in the existing approaches lies in their constraint of processing only a single sketch query at a time (see Figure 1.1). This limitation presents a substantial obstacle for various downstream applications, particularly for users interested in pinpointing individual instances of objects within images.

1.4.2 Cannnot retrieve complex scene based queries with spatial alignment

Current methods accept a single sketch input patch and detect all object instances corresponding to that specific patch within the image. However, this approach falls short when users aim to identify a particular instance among multiple objects within the same image. Additionally, when users wish to detect another category of object, they must create and input another sketch patch, requiring a separate run of the model inference code. This constraint limits users to querying only one class/category at a time.

1.4.3 Addressing Data Scarcity Issue in Paired Hand-Drawn Sketch Queries and Annotated Images

Furthermore, a significant challenge faced by previous methods is the scarcity of data. Acquiring paired hand-drawn queries alongside annotated images poses considerable difficulties. Can we explore synthetic data approaches to mitigate this data scarcity problem?

To overcome the above mentioned limitations, we propose an innovative solution: providing users with a canvas where they can simultaneously draw multiple sketches (see Figure 1.2). This canvas-based approach empowers users to precisely define and retrieve individual objects within complex scenes. For instance, in an image featuring two dogs—one on the left and the other on the right—if a user seeks to retrieve only the dog on the right, the canvas method offers a more versatile and user-friendly solution.

The canvas-based approach we propose effectively addresses above mentioned challenges encountered in prior methods:

1. **Spatial Alignment**: This approach can detect objects with precise spatial alignment. For example, in an image featuring two dogs—one on the left and the other on the right—the user can specify the retrieval of only the dog on the right, ensuring accurate localization.

2. **Complex Multi-Sketch Multi-Category Queries**: The canvas-based approach possesses the capability to process complex multi-sketch queries. When a user draws a dog to the left of a person on the canvas, the system can retrieve images that precisely match this specification while preserving their spatial alignment. This helps in sophisticated object retrieval tasks.

3. **Efficiency with Single Model Inference for Multiple Classes**: With the canvas-based approach, a single model inference can detect multiple classes of objects efficiently, reducing the overall computational overhead thereby improving the efficiency of the system.

4. **Synthetic sketches to alleviate data scarcity**: Leveraging synthetic sketches allows us to train the model for a wider range of applications, eliminating the need for collecting human-drawn sketches, which can be a laborious process.

1.5 Central theme of our thesis

Our primary research focus revolves around simplifying the process of querying multiple objects without the need for constant redrawing. We aim to provide users with a canvas, enabling them to sketch complex scenes and subsequently retrieve objects while preserving spatial alignment and fidelity.

In this work, we aim to build upon the research of [24], [102] and [131], enabling end-users to use multiple sketch queries to detect multiple object instances drawn on an input canvas as shown in Figure 1.2.

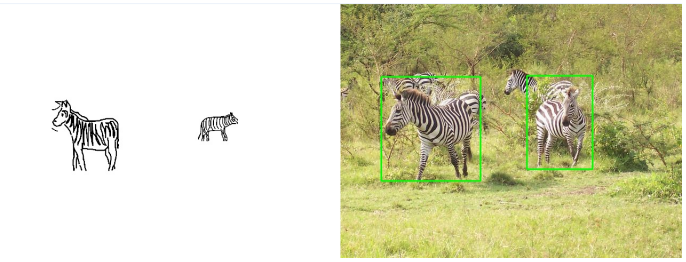

Figure 1.2: Example of object localization using multiple sketch queries on a canvas (left side)

The work by Pau et al. [102] closely resembled our vision to approach this research problem. However, there was no official code implementation available for testing/experimentation. We developed our own version of the Sketch-DETR, which closely resembled the original work proposed by Pau et al [102]. However, after conducting a series of extensive experiments, it became evident that the object query conditioning technique as explained in Section 1.3.2, while intriguing, did not yield significant improvements in our specific context. Consequently, we remove the decoder conditioning and refer to our adapted model as "Sketch Canvas DETR".

In the context of Sketch-Based Image Retrieval (SBIR) and Sketch-Guided Object Localization (SGOL), another significant challenge that has been identified is the scarcity of data, which poses a substantial obstacle for data-dependent cross-modal learning algorithms. Acquiring paired photos and hand-drawn sketches is inherently difficult, limiting the availability of training data and impeding the development of effective cross-modal models for SBIR and SGOL tasks. Furthermore, this data scarcity issue extends to other related tasks, such as object detection [24], where the accuracy of the SBIR model plays a critical role in determining the accuracy of object detection results.

To tackle these issues we propose a novel approach to restructure the existing Sketchy-COCO dataset. We aim to break down scene-level data containing multiple sketch instances into individual images, each featuring only a single sketch instance. To achieve this, we employ a two-fold strategy:

1. **Filtering Background Classes:** We begin by filtering out background classes from the dataset. This step helps in focusing solely on foreground instances, which are the primary objects of interest.

2. **Single Instance Images:** Instead of utilizing images with multiple instances, we ensure that each image contains only a single sketch instance. For instance, if an original image contains two sketch object instances, we create two separate images, each featuring a single instance. This approach serves a dual purpose—it acts as an advanced data augmentation technique and also enhances the model's generalization capabilities.

1.6 Achievements

- Ran baseline experiments for FG-SBIR to check the validity of synthetic sketches

- Successfully modified the DETR code base for sketch guided object detection
- Studied the effect of operators like addition and concatenation when fusing features from sketches and paired images for the pipeline mentioned above
- Developed a cross attention block using single transformer encoder which gave superior results compared to self-attention based methods.

1.7 Overview of Interim Report

This report is structured in the following format:

- Background Theory: We go through basic concepts required to get acquainted with topics related to sketch based image retrieval and object detection
- Literature Review: We briefly go through the existing literature related to the topic of sketch-base image retrieval
- Dataset Exploration: We describe datasets relevant to sketch based applications
- Synthetic Sketch Generation: We go through sketch generation algorithms from natural images. We briefly introduce the readers to the different SOTA methods
- Sketch Canvas DETR: We introduce our custom version of Sketch-DETR and evaluate it's performance on multi-instance object detection
- Future work: We discuss the plan for further improvements and experiments

2 BACKGROUND THEORY

We introduce the readers to some of the background concepts which is required to appreciate the vast field of sketch-based applications. Though numerous papers have been published we describe a select few of the algorithms and papers which will help in understanding the crux of the thesis.

2.1 Vanilla Transformer

The Vanilla Transformer employs an encoder-decoder structure. This architecture processes tokenized input data. Both the encoder and decoder components consist of stacked Transformer layers[135] or blocks, as illustrated in Figure 2.1 .

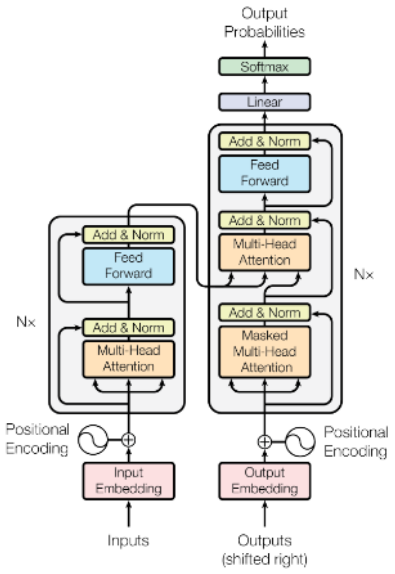

Figure 2.1: Transformer Architecture

Each block comprises two key sub-layers:

1. A multi-head self-attention (MHSA) layer
2. A position-wise fully-connected feed-forward network (FFN) layer.

To facilitate the backpropagation of gradients during training, both the MHSA and FFN layers incorporate a Residual Connection mechanism, denoted as $x \leftarrow f(x) + x$, followed by normalization. This ensures stable training and helps avoid the vanishing gradient problem. Thus, given an input tensor Z, the output of the MHSA and FFN sub-layers [91] can be expressed as follows:

$$Z \leftarrow N(\text{sublayer}(Z) + Z)$$

Here, sublayer$(\cdot)$ represents the mapping implemented by the sub-layer itself, and $N(\cdot)$ denotes normalization, which can take various forms such as Batch Normalization (BN) or Layer Normalization (LN) [148].

2.1.1 Discussion on Normalization

An unresolved issue in Transformer architecture pertains to the order of normalization layers [144], specifically post-normalization versus pre-normalization. The original Vanilla Transformer adopts post-normalization for each MHSA and FFN sub-layer. Resolving this issue can potentially lead to improvements in the efficiency and performance of Transformer-based models.

2.2 Input Tokenization

The Vanilla Transformer, originally designed for machine translation, utilizes tokenized sequences as input. This design was initially intended for machine translation, where source and target sentences are segmented into tokens. Each token can be seen as a node in a graph, allowing the Transformer to model relationships between words effectively. This approach is highly flexible and can be applied to different modalities, making it suitable for a wide range of tasks beyond translation [91].

2.2.1 Special/Customized Tokens

Transformers introduce special/customized tokens as placeholders in token sequences. These tokens serve various semantic purposes and enhance the model's capability. For example, the `[MASK]` token is used in Masked Language Modeling [27], and the `[CLASS]` token aids in classification tasks [35]. These specialized tokens contribute to the Transformer's adaptability, as they allow it to handle diverse tasks with tailored functionalities. [132]

2.2.2 Position Embedding

To retain positional information, position embeddings are added to the token embeddings [27]. The Vanilla Transformer employs sine and cosine functions to generate position embeddings [135], preserving the order of tokens in a sequence. While the specific implementation of position embeddings may vary across different Transformer variants, this feature remains essential for maintaining the sequential context of input data.

2.2.3 Advantages of Input Tokenization

Input tokenization offers several advantages:

1. **Universality and Flexibility:** Modalities, such as text and images, have distinct characteristics and structures. For instance, sentences exhibit sequential patterns suitable for Recurrent Neural Networks (RNNs), while images align with grid matrices, making Convolutional Neural Networks (CNNs) a natural choice. Tokenization allows Transformers to process these modalities universally by representing them as irregular sparse structures. This means that even the Vanilla Transformer can encode multimodal inputs effectively through techniques like concatenation and weighted summation (we have carried out extensive experiments based on such fusion techniques in [refer to exeriments section]), without requiring extensive multimodal modifications.

2. **Flexible Information Organization:** Input tokenization provides flexibility in organizing input information. The Vanilla Transformer introduces temporal information by incorporating position embeddings, ensuring that the model understands the order of tokens in a sequence. This flexibility extends to various tasks. For instance, when applying Transformers to tasks like freehand sketch drawing [146], each input token can encompass diverse drawing stroke patterns, including stroke coordinates, stroke ordering, and pen state (start/end). This adaptability allows Transformers to excel in tasks with heterogeneous data.

3. **Task-Specific Customization:** Input tokenization is compatible with task-specific customized tokens. These tokens are tailored to meet specific task requirements and semantics. For example as stated bfore, the `[MASK]` token is used for Masked Language Modeling, where the model predicts masked words within a sentence. Similarly, the `[CLASS]` token is employed in classification tasks to distinguish between different classes or categories.

This customization ensures that Transformers can be applied to a wide array of tasks while maintaining their effectiveness and efficiency.

2.3 Self-Attention

The core component of Vanilla Transformer is the Self-Attention (SA) operation, also termed "Scaled Dot-Product Attention". Let $X = [x_1, x_2, \ldots] \in \mathbb{R}^{N \times d}$ be an input sequence of N elements/tokens, and an optional preprocessing step involves positional encoding, either by point-wise summation $Z \leftarrow X \oplus \text{PositionEmbedding}$ or concatenation $Z \leftarrow \text{concat}(X, \text{PositionEmbedding})$.

The Self-Attention (SA) operation proceeds as follows: After preprocessing, the embedding Z undergoes three projection matrices $W_Q \in \mathbb{R}^{d \times d_q}$, $W_K \in \mathbb{R}^{d \times d_k}$, and $W_V \in \mathbb{R}^{d \times d_v}$, where $d_q = d_k$, to generate three embeddings: Q (Query), K (Key), and V (Value):

$$Q = ZW_Q, \quad K = ZW_K, \quad V = ZW_V.$$

The output of the self-attention operation is defined as:

$$Z = \text{SA}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_q}}\right)V.$$

Here, Softmax computes the softmax scores for the dot products of Q and K, scaled by $\sqrt{d_q}$, and then uses these scores to weight the V embeddings to produce the final output Z [91].

2.4 Masked Self-Attention (MSA)

In practice, modifications to the self-attention mechanism are often necessary to aid the Transformer decoder in learning contextual dependencies and to prevent positions from attending to subsequent positions. This modification is achieved by introducing a masking matrix, resulting in the modified Self-Attention (MSA) operation:

$$Z = \text{MSA}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_q} \cdot M}\right)V,$$

where M is a masking matrix. For example, in models like GPT [93] , an upper triangular mask is applied to enable look-ahead attention, ensuring that each token can only attend to previous tokens and not future ones.

2.5 Multi-Head Self-Attention

In practice, it's common to stack multiple self-attention sub-layers in parallel, and their concatenated outputs are combined through a projection matrix W to create a structure known as Multi-Head Self-Attention [135]:

$$Z = \text{MHSA}(Q, K, V) = \text{concat}(Z_1, \ldots, Z_H)W,$$

where each head $Z_h = \text{SA}(Q_h, K_h, V_h)$ for $h \in [1, H]$, and W represents a linear projection matrix. The concept behind Multi-Head Self-Attention (MHSA) is a form of ensemble learning. MHSA allows the model to jointly focus on information from multiple sub-spaces of representation, enhancing its ability to capture complex relationships.

2.6 Feed-Forward Network

The output of the multi-head attention sub-layer is processed by a position-wise Feed-Forward Network (FFN) composed of successive linear layers with a non-linear activation function. For example, a two-layer FFN can be represented as:

$$\text{FFN}(Z) = \sigma(ZW_1 + b_1)W_2 + b_2,$$

where W_1, b_1, W_2, and b_2 represent the weights and biases of the two linear transformations, and $\sigma(\cdot)$ is a non-linear activation function, such as ReLU(·) [171] or GELU(·) [172]. In some Transformer literature, the term Multi-Layer Perceptron (MLP) is also used to refer to FFN.

2.7 Vision Transformer

The Vision Transformer (ViT) [35] introduces an image-specific input pipeline, where the input image needs to be divided into fixed-size patches (e.g., 16x16, 32x32). After linear embedding and the addition of position embeddings, these patch-wise sequences are encoded by a standard Transformer encoder.

Given an image $X \in \mathbb{R}^{H \times W \times C}$ (where H represents height, W width, and C channels), ViT reshapes X into a sequence of flattened 2D patches: $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$, where $(P \times P)$ is the patch resolution, and $N = \frac{H \cdot W}{P^2}$.

To enable classification, a common approach is to prepend an additional learnable embedding called the "classification token" [CLASS] to the sequence of embedded patches [91] :

$$Z \leftarrow \text{concat}([\text{CLASS}], XW),$$

where W represents the projection.

2.8 Multimodal Learning

In various computer vision tasks, different modalities, such as natural language, image or audio signals, often contain complementary information while also overlapping in their representation of a common concept. The field of multimodal learning aims to leverage this complementary information to enhance the performance of these tasks. One crucial aspect of multimodal learning is the exploration of efficient methods for fusing information from multiple modalities.

2.8.1 Existing Fusion Methods

Simple fusion methods like concatenation and element-wise summation have been extensively studied in previous research [91]. These methods involve simply combining the features or representations from different modalities.

2.8.2 Tensor Fusion

To facilitate more efficient cross-modality interaction, Zadeh et al. introduced a tensor fusion mechanism [156]. This approach seeks to capture the relationships between modalities by working with tensors, allowing for richer representations of multimodal data.

2.8.3 Low-Rank Fusion

In response to the computational challenges associated with tensor fusion, an efficient low-rank fusion technique has been proposed [83]. This method is designed to address the exponential growth in dimensionality that occurs when applying tensor fusion to multimodal data.

2.9 Combining Modalities in Multimodal Deep Learning (Multimodal Fusion)

In the realm of multimodal Transformers, cross-modal interactions, such as fusion and alignment, are primarily processed through self-attention or cross-attention mechanisms and their variations. In this section, we review various multimodal modeling practices within Transformers, emphasizing mainly self-attention designs [91]. We focus on 6 types of fusion techniques (see Figure 2.2):

1. Early Summation
2. Early Concatenation
3. Hierarchical Attention (Multi-Stream to One-Stream)
4. Hierarchical Attention (One-Stream to Multi-Stream)
5. Cross-Attention
6. Cross-Attention to Concatenation

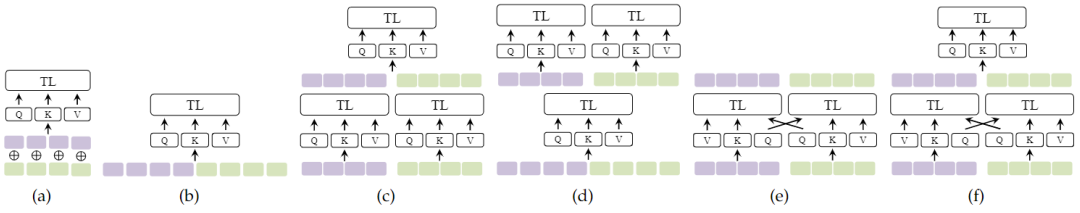

Figure 2.2: Transformer-based cross-modal interactions: (a) Early Summation, (b) Early Concatenation, (c) Hierarchical Attention (multi-stream to one-stream), (d) Hierarchical Attention (one-stream to multi-stream), (e) Cross-Attention, and (f) Cross-Attention to Concatenation. "Q": Query embedding; "K": Key embedding; "V": Value embedding. "TL": Transformer Layer.

Given inputs X_A and X_B from two arbitrary modalities, Z_A and Z_B represent their respective token embeddings. Let Z denote the token embedding sequence produced as a result of the multimodal interactions. The function $Tf(\cdot)$ indicates the processing performed by Transformer layers/blocks.

2.9.1 Early Summation

In this approach (see Figure 2.3), token embeddings from different modalities are linearly combined with weights and then processed by Transformer layers.

$$Z = \text{Tf}(\alpha Z(A) + \beta Z(B)) \tag{2.1}$$

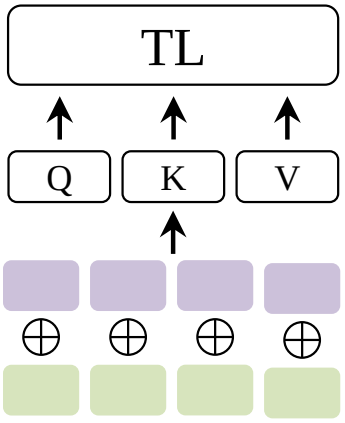

Figure 2.3: Early Summation

2.9.2 Early Concatenation

Token embedding sequences from different modalities are concatenated into a single sequence and fed into Transformer layers (see Figure 2.4).

$$Z = \text{Tf}(C(Z(A), Z(B))) \tag{2.2}$$

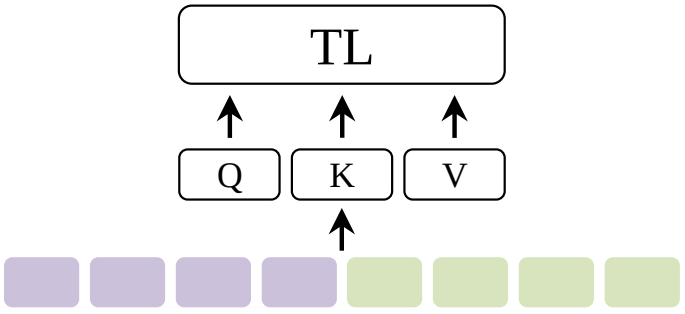

Figure 2.4: Early Concatenation

2.9.3 Hierarchical Attention (Multi-Stream to One-Stream)

Multimodal inputs are processed independently by separate Transformer streams, and their outputs are concatenated and further fused by another Transformer (see Figure 2.5).

$$Z = \text{Tf}_3(C(\text{Tf}_1(Z(A)), \text{Tf}_2(Z(B)))) \tag{2.3}$$

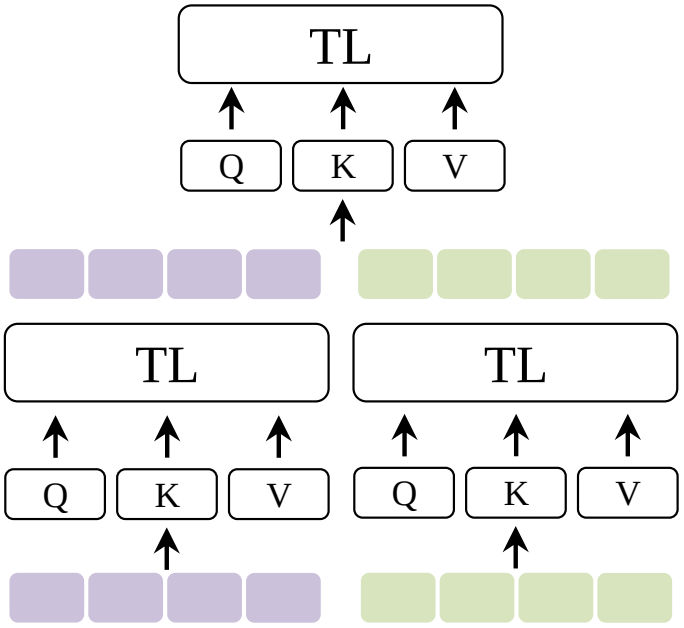

Figure 2.5: Hierarchical Attention (Multi-Stream to One-Stream)

2.9.4 Hierarchical Attention (One-Stream to Multi-Stream)

This approach involves encoding concatenated multimodal inputs using a shared single-stream Transformer, followed by separate Transformer streams for each modality (see Figure 2.6).

$$C(Z(A), Z(B)) \leftarrow \text{Tf}_1(C(Z(A), Z(B))) \tag{2.4}$$

$$Z(A) \leftarrow \text{Tf}_2(Z(A)) \tag{2.5}$$

$$Z(B) \leftarrow \text{Tf}_3(Z(B)) \tag{2.6}$$

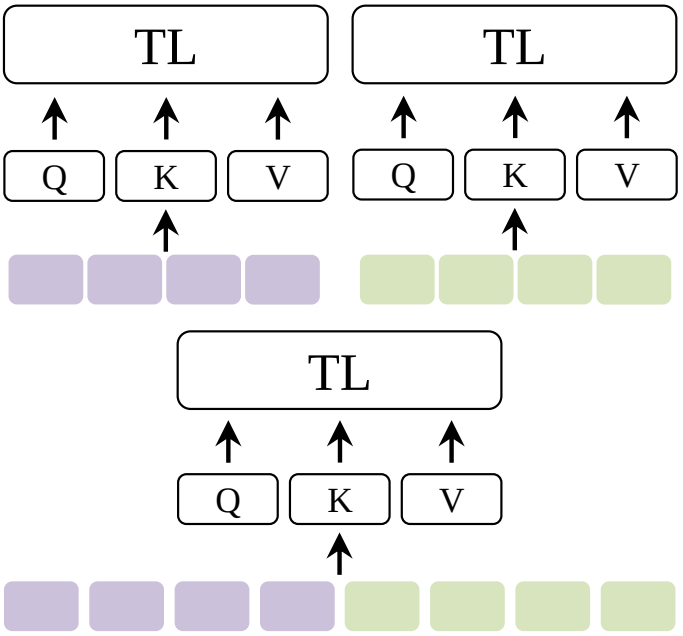

Figure 2.6: Hierarchical Attention (One-Stream to Multi-Stream)

2.9.5 Cross-Attention

Cross-attention is applied to two-stream Transformers, where Query (Q) embeddings are exchanged between modalities, allowing cross-modal interactions (see Figure 2.7).

$$Z(A) \leftarrow \text{MHSA}(Q(B), K(A), V(A)) \tag{2.7}$$

$$Z(B) \leftarrow \text{MHSA}(Q(A), K(B), V(B)) \tag{2.8}$$

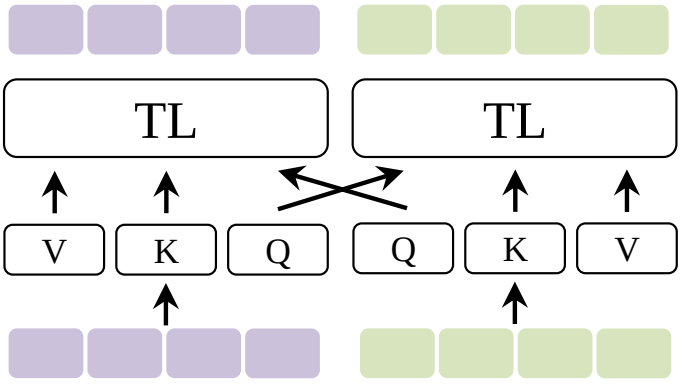

Figure 2.7: Cross-Attention

2.9.6 Cross-Attention to Concatenation

Two-stream cross-attention outputs are concatenated and further processed by another Transformer layer to capture global cross-modal context (see Figure 2.8).

$$Z(A) \leftarrow \text{MHSA}(Q(B), K(A), V(A)) \tag{2.9}$$

$$Z(B) \leftarrow \text{MHSA}(Q(A), K(B), V(B)) \tag{2.10}$$

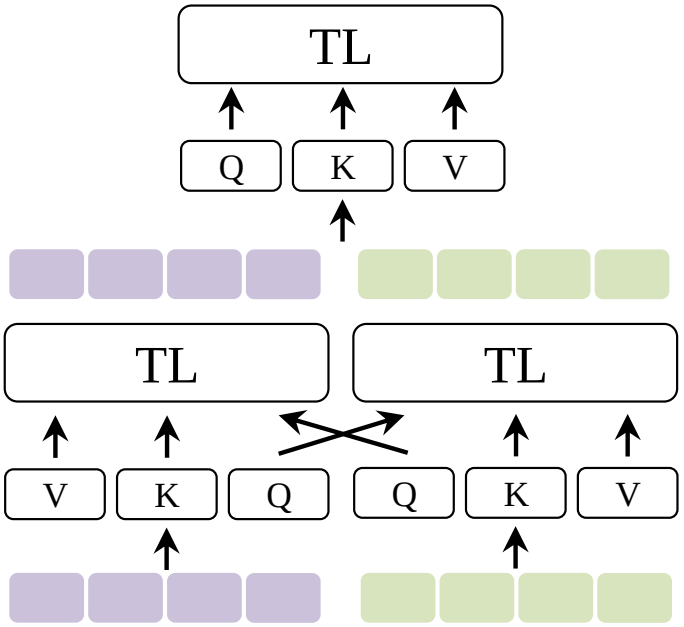

Figure 2.8: Cross-Attention to Concatenation

2.9.7 Discussion on complexity

Hierarchical Attention and Cross-Attention mechanisms can be superior to simpler techniques like addition and concatenation. However, they come at a cost of larger memory footprints since multiple transformer encoders (which in turn have multiple transformer attention layers) would be employed.

2.10 Sketch-Based Image Retrieval (SBIR) vs Fine-Grained Sketch-Based Image Retrieval (FG-SBIR)

SBIR aims to retrieve natural images that match a given hand-drawn sketch, regardless of the level of detail or realism of the sketch. In other words, SBIR does not focus on fine-grained visual

differences between objects but only on the class label. In such cases, text is often a simpler form of input when only category-level retrieval is required [24]

FG-SBIR, on the other hand, is a more challenging task that aims to retrieve natural images that match a given hand-drawn sketch with fine-grained details, such as specific texture, color, or shape features of an object. FG-SBIR requires more detailed sketches and a deeper understanding of the visual characteristics of objects [6].

2.11 CLIP (Contrastive Language-Image Pre-Training)

CLIP (Contrastive Language-Image Pre-Training) is a neural network model developed by OpenAI that can learn visual concepts through supervised learning from natural language. It can perform tasks such as image classification, object detection, and image retrieval using only text descriptions as supervision [94]

The model is trained on a massive dataset of 400M image-text pairs, where it learns to map natural language descriptions to corresponding images by maximizing the similarity between them. This is achieved by using a contrastive loss function that encourages the model to correctly associate a given text description with its corresponding image, while at the same time minimizing the similarity between the image and other irrelevant text descriptions.

The CLIP model has achieved state-of-the-art results on a variety of visual tasks, including image classification and natural language-based image retrieval, without the need for task-specific fine-tuning.

2.11.1 Prompt Learning

Prompt learning [24] is a process in which the prompt template and the associated class-specific weights used in vision-language models like CLIP and ALIGN are learned directly from data, rather than being manually designed or pre-specified.

A prompt template is a predefined text pattern that is used as input to a text encoder in such models. The template includes special tokens that are replaced with specific information, such as the name of a class, to generate class-specific weights for classification.

For example, a prompt template for classifying images of different fruits could be ”a photo of a fruit, a type of food”. The ”fruit” token is a placeholder that is replaced with the name of the actual fruit, such as ”apple” or ”banana”, to generate class-specific weights for classification.

Prompt templates can be customized for different image classification tasks and allow for more flexible and efficient training of vision-language models. They enable the same text encoder and classification weights to be used across different classes and tasks, by simply swapping out the specific information in the template. This approach is particularly useful for open-set classification tasks, where the model needs to be able to recognize new classes that were not included in the training data.

2.11.2 Need for CLIP

ResNet and other similar image classification models are excellent at top-one or top-five classification accuracy, but they have limitations. For example, they may fail in robustness tests or when presented with adversarial examples or slight differences in image distribution. In contrast, CLIP uses a contrastive learning approach that unifies images and text and turns image classification into a text similarity problem. CLIP performs much better on different kinds of datasets, making it more versatile than ResNet. Additionally, building large labeled datasets like ImageNet is expensive and time-consuming, whereas CLIP can learn representations from unlabeled data using contrastive pre-training. Furthermore, ResNet is limited to the 1000 ImageNet categories, while CLIP can generalize to new categories using zero-shot learning.

2.12 Object detection

Object detection techniques can be classified into different types based on their approaches and methodologies. The primary distinction lies in their fundamental substrate, i.e., the underlying layer upon which predictions are made. The classic approach and the newer approach as shown by DETR offer two distinct perspectives on tackling the problem of object detection.

2.12.1 Classical Approach

1. **Classical Approach:**

   This approach treats object detection as a machine learning problem and models it as a classification task. It involves several interconnected steps:

- **Quantization and Regression:** To address the challenge of too many boxes for classification, a proxy problem is introduced. Boxes are quantized into a finite set of representative boxes, and a model is trained to predict the quantization error. This introduces both classification and regression tasks.
- **Label Assignment Heuristics:** Since quantized boxes do not exactly match ground truth, labeling heuristics based on criteria like intersection over union (IoU) thresholds are used to determine foreground and background boxes.
- **Redundancy Removal:** The independent predictions for each box can lead to redundant detections. Non-maximum suppression (NMS) is applied to remove redundant detections by selecting boxes with the highest confidence and suppressing overlapping detections.
- **Imbalanced Data Handling:** Handling imbalanced data is essential, as foreground boxes are significantly fewer than background boxes. Techniques like focal loss, cascading classifiers, and hard negative mining are employed to mitigate this imbalance.

2.12.2 Supervised object detection

Object detection is the task of identifying and classifying objects in an image. There are two main categories of object localization methods: proposal-free and proposal-based. Proposal-free methods are faster during inference but often have lower performance compared to proposal-based methods. Proposal-based methods first generate object proposals and then refine them by classifying them into object categories. Previous methods used selective search to generate proposals, but the two stages were trained independently. Faster R-CNN introduced a region proposal network (RPN) that made the detection pipeline end-to-end trainable. Though many other algorithms have been proposed since, Faster-RCNN has been robust in accuracy and is still used today in research and production. Despite the introduction of various other object detection algorithms, Faster-RCNN has remained robust in terms of accuracy and continues to be used in both research and production.

2.12.2.1 Faster R-CNN Object Detection

Faster R-CNN is a popular object detection framework that consists of two main components: a Region Proposal Network (RPN) and a Fast R-CNN detector. The RPN generates proposals, which are regions of interest (RoIs) that may contain objects, while the Fast R-CNN detector takes the RoIs as input and performs object classification and bounding box regression.

The RPN is implemented as a fully convolutional network that shares convolutional layers with the Fast R-CNN detector. It takes an image as input and outputs a set of object proposals, each represented by a bounding box and an objectness score. The RPN uses an anchor-based approach, where a set of anchor boxes are defined at each spatial location, and the RPN predicts offsets and scores for each anchor box. The anchor boxes have different aspect ratios and scales to handle objects of different sizes and shapes.

The output of the RPN is a set of proposals that are passed to the Fast R-CNN detector. The Fast R-CNN detector takes each proposal as input and performs feature extraction using a region of interest pooling layer. The features are then fed into a series of fully connected layers for classification and regression. The final output of the detector is a set of class probabilities and bounding box offsets for each proposal.

The loss function used to train the Faster R-CNN network is a combination of the RPN loss and the Fast R-CNN loss. The RPN loss consists of a classification loss and a bounding box regression loss, while the Fast R-CNN loss consists of a classification loss and a bounding box regression loss. The total loss is a weighted sum of the two losses.

$$\mathcal{L}_{FasterR-CNN} = \mathcal{L}_{RPN} + \mathcal{L}_{FastR-CNN}$$

where $\mathcal{L}_{RPN}$ and $\mathcal{L}_{FastR-CNN}$ are the RPN loss and Fast R-CNN loss, respectively.

2.12.3 Weakly supervised object detection

Weakly supervised object detection refers to a type of computer vision task where the goal is to detect objects in an image using only partial or incomplete supervision, as opposed to fully supervised methods that require precise bounding box annotations for every object in every image. In weakly supervised object detection, the training data may only include image-level labels, indicating the presence or absence of an object in an image, or may include noisy or incomplete bounding box annotations [115].

There are different approaches to weakly supervised object detection, but one common strategy is to use a combination of localization and classification methods to identify the location and category of objects in an image. For example, some methods may use saliency maps or attention

mechanisms to highlight regions of an image that are likely to contain an object, and then apply classification models to these regions to identify the object category.

2.12.4 Extremely weakly supervised object detection (EWSOD)

Recently, Pinaki et al [24] introduced extremely weakly supervised object detection (EWSOD) that does away with the need for any supervised labels at all .

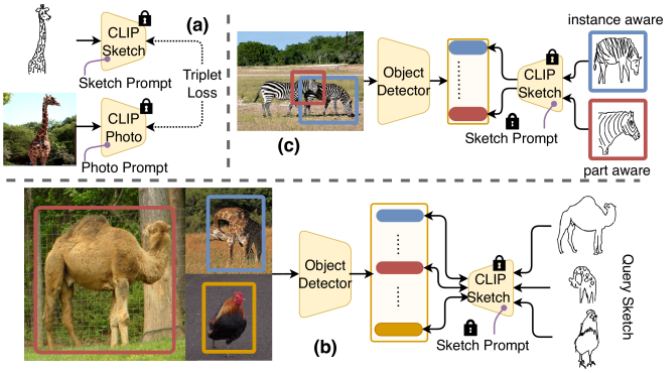

Figure 2.9: Extremely weakly supervised object detection

EWSOD is a technique for training object detectors with minimal annotation requirements. Instead of using bounding box annotations, it relies on image-level class labels, which indicate whether objects of certain classes are present in the image or not. To avoid the use of bounding box annotations, EWSOD employs Pre-trained Region Proposal Network (RPN), Heuristic-based selective search or Edge boxes to generate box proposals. For each proposed region r_i, patch features f_r are extracted, and these features are split into a classification head $x_c = \phi_{cls}(f_r) \in \mathbb{R}^{R\times(|C|+1)}$ and a detection head $x_d = \phi_{det}(f_r) \in \mathbb{R}^{R\times(|C|+1)}$. The classification head ϕ_{cls} assigns scores to individual proposals for each of the C classes and a background class (represented as $|C|+1$) using softmax as follows:

$$\sigma_{cls}(x^c_{i,j}) = \frac{e^{x^c_{i,j}}}{\sum_{k=1}^{|C|+1} e^{x^c_{i,k}}} \tag{2.11}$$

The detection head ϕ_{det} measures the contribution of each patch i to being classified into class j (including the background class) using softmax across all R regions:

$$\sigma_{det}(x_{i,j}^d) = \frac{e^{x_{i,j}^d}}{\sum_{k=1}^{R} e^{x_{k,j}^d}} \tag{2.12}$$

Training is performed using image-level labels $Y = [y_0, y_1, \ldots, y_{|C|}]^T \in \mathbb{R}^{(|C|+1)\times 1}$, where $y_c = 1$ or 0, indicating whether an instance of class c is present in the image or not. The combined score ω_0 is computed as an element-wise product of the class scores σ_{cls} and the patch scores σ_{det}, and it is used to estimate the probability of instances from class c being present in the image:

$$\hat{y}_c = \sum_{i=1}^{R} \omega_{i,c}^0 \tag{2.13}$$

Training is conducted using multi-class cross-entropy loss:

$$L_{ws} = -\sum_{c=1}^{|C|+1} (y_c \log \hat{y}_c + (1 - y_c) \log(1 - \hat{y}_c)) \tag{2.14}$$

EWSOD differs from standard Supervised Object Detection (SOD) in that it only uses image-level class labels for training, without the use of bounding box annotations. To refine the proposals iteratively, an iterative refinement classifier $\omega_k = \phi_{*cls}(f_r)$ is introduced. This classifier is supervised using pseudo scores $l^{(k-1)}$ from the $(k-1)$th iteration. The steps for pseudo score assignment involve selecting the patches with the highest scores for each class, assigning regions with high overlap to the corresponding class, and assigning regions with low overlap to the background class.

The refinement loss is defined as:

$$L_{k_{\text{ref}}} = \frac{1}{R} \sum_{i=1}^{R} \sum_{c=1}^{|C|} \omega_{i,j}^{(k-1)} l_{i,j}^{(k-1)} \log \omega_{i,j}^k \tag{2.15}$$

Both EWSOD and SOD are constrained to detect objects belonging to a predefined set of C classes.

2.12.5 DETR Approach

DETR represents a newer perspective on object detection, focusing on direct set-level prediction and interaction. It differentiates itself from the classic approach in several ways:

1. **Fundamental Substrate:** DETR employs a set of learned object queries, which are not constrained by prior geometric meaning. These queries interact with image features through a transformer decoder.
2. **Direct Set-Level Prediction:** Instead of predicting classifications and regressions for quantized boxes, DETR predicts categories and bounding boxes directly for each object query. This eliminates the need for label assignment heuristics and quantization error prediction.
3. **Non-Maximum Suppression Integration:** The transformer decoder in DETR allows for set-level interactions, enabling learned non-maximum suppression directly in the model.
4. **Permutation Invariance:** The transformer used in DETR operates on permutations, making it invariant to the order of object queries, enhancing its ability to handle set-level interactions.

2.13 Evaluation Metric - Average Precision at IoU 0.5 ($AP\_0.5$)

We use a metric called as Average Precision at IoU 0.5 ($AP\_0.5$)

$AP\_0.5$ is a common evaluation metric used in object detection and image segmentation tasks to measure the accuracy of a model's predictions. It specifically assesses the precision of object localization.

- **Average Precision (AP)**: Average Precision is a metric used to evaluate the precision-recall curve of a model. It measures how well a model can identify objects or regions of interest within an image while considering different confidence thresholds. The precision-recall curve illustrates the trade-off between precision (the ratio of true positives to all positive predictions) and recall (the ratio of true positives to all actual positives) at various confidence thresholds.
- **IoU (Intersection over Union)**: IoU is a measure of the overlap between the predicted bounding box (or region) and the ground truth bounding box. It is calculated as the ratio of

the area of overlap between the predicted and ground truth boxes to the area of their union. In the context of $AP\_0.5$, IoU is set to 0.5, which means that the predicted bounding box is considered correct if it has an overlap of at least 50% with the ground truth bounding box.

So, when we refer to *Average Precision at IoU 0.5* ($AP\_0.5$), it means that we are computing the average precision of a model's object localization predictions while using a minimum IoU threshold of 0.5 (50% overlap) to determine whether a predicted bounding box is correct. $AP\_0.5$ provides a measure of how effectively a model can accurately localize objects in an image with a relatively lenient criterion for correctness.

3 LITERATURE REVIEW

3.1 Sketch-guided object localization

Sketch-guided localization can be used to detect and localize objects in an image by providing a sketch query. This task is distinctively different from the traditional sketch-based image retrieval task where the gallery set often contains images with only one object [130]. Despite its potential significance, sketch-guided localization has received limited research attention to date. Existing work makes use of a supervised learning approach using cross-modal attention from deep learning models trained on natural images and sketches to generate an attention matrix that can guide a region proposal network (RPN) to localize objects [130]. However, this work employs a very complicated method of cross modal attention described in next section.

Cross-modal Attention for Query-guided Object Proposal Generation (see Figure 3.1) is an attention module that leverages information from sketch queries, in addition to the image data.

The core idea is to establish a connection between the query or sketch and the image, allowing the system to focus on regions in the image that are most relevant to the given query. This is achieved through the use of attention mechanisms, which assign different levels of importance to different parts of the image based on their similarity or relevance to the query.

The cross-modal attention can effectively propose objects that are low in resolution, partially obscured, or hidden within a cluttered scene. This is achieved by taking into account both the content of the image and the additional information provided by the query or sketch.

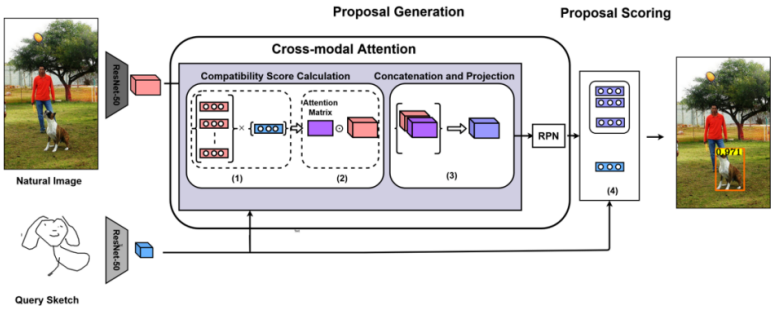

Figure 3.1: Cross-modal Attention for Query-guided Object Proposal Generation

3.2 Sketch-DETR: Enhancing Object Localization with Sketch Conditioning

Sketch-DETR [102] introduces an approach to object localization by leveraging sketches as conditioning cues within the DETR (DEtection TRansformer) architecture. Two distinct conditioning methods are explored (see Figure 3.2):

3.2.1 Object Query Conditioning

In the first method, sketch features are incorporated at the object query level. This involves feeding sketch features, obtained through a CNN backbone denoted as $\zeta(\cdot)$, into the transformer decoder alongside query objects. The primary motivation behind this scheme is to provide query objects with both spatial information and content details. This is achieved through a simple linear layer that concatenates query objects and sketch features, resulting in a new tensor $\widetilde{q}_i$ of shape d dimensions.

3.2.2 Encoder Concatenation Conditioning

The second conditioning method involves combining features obtained by the DETR CNN backbone ($f \in \mathbb{R}^{d \times H \times W}$) with sketch features ($f_s \in \mathbb{R}^d$) extracted by the sketch CNN backbone. This concatenation is performed, with f_s repeated to match the size of f. Subsequently, a 1×1 convolution reduces the dimensions to $d \times H \times W$. This approach aims to enhance features in regions highly correlated with the provided sketch. Both conditioning methods are depicted in Figure 2.

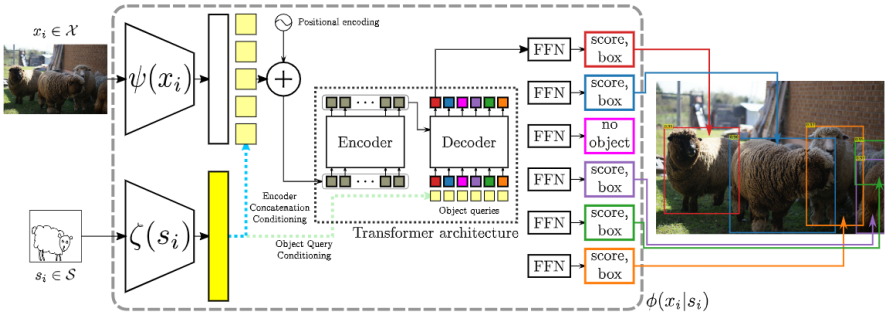

Figure 3.2: Sketch DETR

3.2.3 Learning Objectives

The learning objectives follow a scheme proposed by Carion et al. [16], tailored to the binary case of object presence or absence. Proper matching between the set of N predicted objects and the ground-truth set is crucial. A bipartite matching minimizes the matching cost, considering

predicted boxes and objectness scores. The Hungarian loss is then computed based on assigned pairs. It incorporates class probabilities and a combination of Generalized Intersection over Union (IoU) and L1 loss for predicted boxes.

Optional losses for instance segmentation mask prediction include the Focal Loss to address data imbalance and the DICE/F-1 loss, optimizing the Dice coefficient for predicted masks.

3.3 Synthetic sketch generation techniques

3.3.1 CLIPascene

CLIPascene [137] is a method proposed by Yael Vinker et al. for generating sketches from scene images with different levels of abstraction. The method employs two types of abstraction: fidelity and visual simplicity, and allows users to select the desired level of abstraction based on their personal preferences. The approach involves training two multi-layer perceptron (MLP) networks to learn stroke placement and removal while preserving the recognizability and semantics of the sketch. The proposed method is capable of generating sketches of complex scenes, including those with complex backgrounds and subjects. The paper provides a project page for further exploration and discusses the application of CLIPascene in computer vision and graphics.

3.3.2 Learning to generate line drawings that convey geometry and semantics

Caroline Chan et al [18] presents an unpaired method for creating line drawings from photographs. Current methods often rely on high quality paired datasets to generate line drawings. However, these datasets often have limitations due to the subjects of the drawings belonging to a specific domain, or in the amount of data collected. Although recent work in unsupervised image-to-image translation has shown much progress, the latest methods still struggle to generate compelling line drawings. The authors observe that line drawings are encodings of scene information and seek to convey 3D shape and semantic meaning. They build these observations into a set of objectives and train an image translation to map photographs into line drawings. A geometry loss which predicts depth information from the image features of a line drawing, and a semantic loss which matches the CLIP features of a line drawing with its corresponding photograph is used. Few examples see Figure 3.7

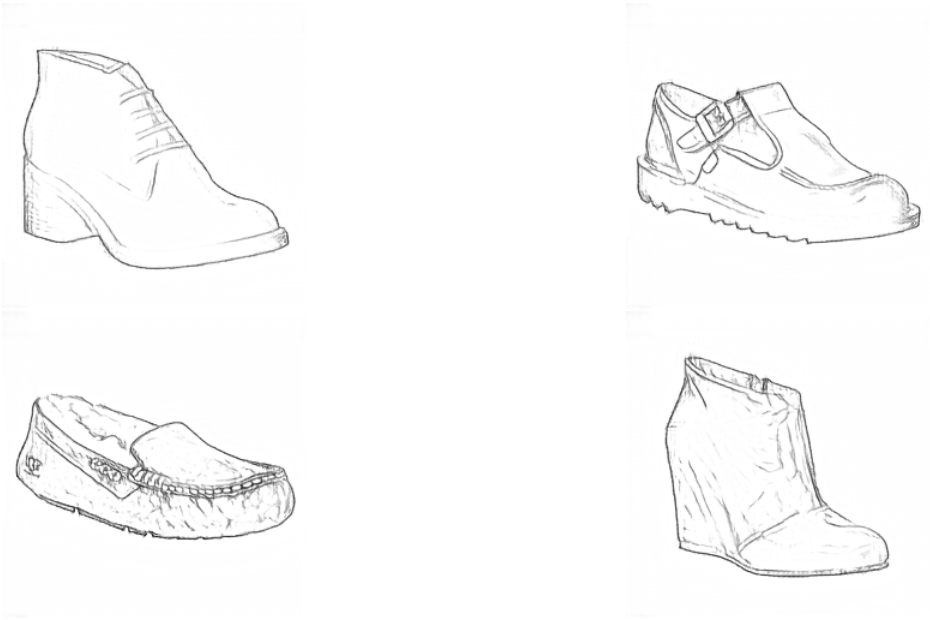

Figure 3.7: Caroline Sketch Samples

3.3.3 CLIPasso

CLIPasso [136] is a method for generating object sketches with different levels of abstraction, guided by geometric and semantic simplifications. The method is based on CLIP (Contrastive-Language-Image-Pretraining), which has the ability to distill semantic concepts from sketches and images. CLIPasso defines a sketch as a set of Bézier curves and uses a differentiable rasterizer to optimize the parameters of the curves directly with respect to a CLIP-based perceptual loss. The degree of abstraction is controlled by varying the number of strokes. The generated sketches demonstrate multiple levels of abstraction while maintaining recognizability, underlying structure, and essential visual components of the subject drawn. CLIPasso leverages the semantic understanding of CLIP to generate sketches.

The sketch generation process at test time is shown in Figure 3.12.These images were generated with a stroke size of 25. As shown, the expressive power of CLIPasso is not that great when it comes to recreating the complex floral designs. More number of strokes would be required which would consume more time. It takes around 42 minutes to generate a single sketch.

Figure 3.12: Clipasso sketches generated at iterations 0, 240, 600 and 1000

4 DATASET EXPLORATION

4.1 Sketchy:

The Sketchy database [111] was presented as the first collection of sketch-photo pairs. Crowd workers were asked to sketch photographic objects from 125 categories resulting in 75,471 sketches of 12,500 objects. Fine-grained associations between photos and sketches were established using this database to train cross-domain convolutional networks. The learned representation outperformed hand-crafted and deep features trained for sketch or photo classification. The Sketchy database was used as a benchmark for fine-grained retrieval. An important point to be noted is that the participants were not allowed to trace photos, but instead, photos were revealed and then hidden, requiring them to sketch from memory. This approach is similar to the way a user of a sketch-based image retrieval system would draw based on a mental image of a desired object. Hence, this dataset can be used to simulate real-life image retrieval scenarios.

4.2 QuickDraw-Extended Dataset:

The QuickDraw-Extended dataset [31] was created using the Google Quick, Draw! dataset, which consists of 50 million drawings across 345 categories. The Quick, Draw! game asks users to draw sketches of a given category while the computer tries to classify them. A subset of 110 categories was selected, with 80 for training and 30 for testing. Categories such as circle or zigzag were discarded as they cannot be used for appropriate SBIR. Images were extracted from Flickr and tagged with the corresponding label to serve as the retrieval gallery. Manual filtering was performed to remove outliers. Additionally, a test split was provided to ensure that test classes were not present in ImageNet when using pre-trained models. The final dataset contains 330,000 sketches and 204,000 photos and is designed for large-scale Zero-Shot Sketch-Based Image Retrieval (ZS-SBIR). The dataset addresses the large domain gap between non-expert drawers and photos that is not considered in previous benchmarks and is expected to provide better insights into the real performance of ZS-SBIR in a real scenario [131]

4.3 SketchyCOCO Dataset

The SketchyCOCO dataset is a comprehensive resource introduced in the context of the CVPR paper ”SketchyCOCO: Image Generation from Freehand Scene Sketches.” It is primarily focused

on the generation of entire scenes from freehand sketches, with an emphasis on complex and complete scene-level image synthesis.

4.3.1 Organization of dataset

- **Object-level Data**: It contains triplets of foreground sketches, foreground images, and foreground edge maps covering 14 different object classes. It also includes pairs of background sketches and background images covering 3 classes.
- **Scene-level Data**: This section comprises pairs of foreground images combined with background sketches to create complete scene images. Additionally, there are pairs of scene sketches and scene images, along with segmentation ground truth data for scene sketches.

The dataset has been augmented with additional images and corresponding data for specific object classes.

SketchyCOCO and the Sketchy database differ in their primary purposes. SketchyCOCO is tailored for scene-level image generation, while the Sketchy database is geared towards object recognition and retrieval tasks involving sketches.

5 METHODOLOGY

5.1 Problem formulation

Consider two sets, denoted as $\mathcal{P}$ and $\mathcal{S}$, representing photos and sketches of the same size respectively. Each pair of a photo and its corresponding sketch canvas share a common characteristic: both contain a set of n bounding boxes, each representing an individual object of interest belonging to C categories. The objects within the sketch canvas and photo are spatially aligned and have a one to one correspondence thought there maybe a difference in their sizes.

Mathematically, let $\mathcal{P} = \{p_i\}_{i=1}^{N}$ represent the set of N photos, and $\mathcal{S} = \{s_i\}_{i=1}^{N}$ denote the set of corresponding sketches. For each sketch instance s_i within the sketch canvas containing total n instances there exists a corresponding bounding box b_i within the photo $\{b_j\}_{j=1}^{n}$

At inference, the goal is to detect bounding boxes for the n sketch instances present in the query canvas.

5.2 DETR

Building upon the concept of DETR (DEtection TRansformer) introduced by Carion et al. [16], we present a variant called Sketch-Guided-DETR (SG-DETR) to address the challenge of Sketch-Guided Object Localization (SGOL) [102].

The DETR architecture [16] comprises several key components, including a Convolutional Neural Network (CNN) backbone, an encoder-decoder transformer, and feed-forward networks (FFNs) that contribute to generating final predictions. A unique aspect of DETR's design is the application of a set-based bipartite matching loss, which enforces distinct predictions for each ground-truth bounding box. DETR's remarkable performance in comparison to other object detectors can be attributed to its inherent capability to reason about object relationships, facilitated by the self-attention mechanism embedded in the transformer architecture.

5.3 Sketch Canvas DETR (SC-DETR)

Our model referred to as Sketch Canvas DETR (SC-DETR) (see Figure 5.1) is an extension of the DETR framework [16], incorporating specific modifications tailored to address the task of

detecting objects with spatial alignment to sketch queries drawn on a canvas. The following modifications have been implemented:

- The DETR module is extended to accommodate two distinct modalities: photographs and sketches. Each modality undergoes processing through a ResNet-50 feature extractor. The output representation derived from the final layer of the feature extractor is then fed into the core DETR module.
- In place of the original encoder in the DETR architecture, a cross-attention encoder block is introduced. This encoder block operates independently on each modality, with the hidden representation in corresponding encoder layers being compared with the original sketch embedding, allowing them to be processed separately and enabling more effective integration of the spatial information inherent to sketch queries.
- Class labels are converted to only 2 classes background and no_ background. This simple technique helps in achieving better generalization and can support unseen classes at inference.

We begin by extracting features from the input images and sketches. The image features have a dimension of $batch\_size \times 3 \times height \times width$, while sketch features are of size $batch\_size \times 1 \times height \times width$. We utilize a ResNet-based architecture to effectively extract image and sketch features. This results in features of dimensions $batch\_size \times 512 \times fm\_height \times fm\_width$, where fm denotes the feature map.

5.3.1 Fusion Strategies

Depending on the fusion strategy chosen, the feature dimensions change accordingly:

- **Elementwise Addition**: Features remain $batch\_size \times 512 \times fm\_height \times fm\_width$.
- **Concatenation**: Features become $batch\_size \times 1024 \times fm\_height \times fm\_width$ as the channel dimension doubles.
- **Cross-Attention with single encoder**: Modalities are passed as is to the query, keys, and values. In subsequent layers, the hidden representations are passed as keys and values whereas the sketch features are passed as queries for every encoder layer.

5.3.2 Input Projection

To prepare the features for the transformer decoder, we use an input projection method that adjusts the dimensions to $batch\_size \times hidden\_dim \times (fm\_height \times fm\_width)$. This is achieved through a 1×1 convolutional layer.

5.3.3 Transformer Encoder

The transformed features are then fed into a transformer encoder consisting of six layers. These encoder layers enhance the representation of the fused features.

5.3.4 Transformer Decoder

The output of the encoder is jointly fed to the transformer decoder, along with query embeddings. Initially, the query embeddings are represented as tensor arrays filled with zeros, with a shape of $num\_queries \times batch\_size \times hidden\_dim$. It is important to note that the first self-attention mechanism within the decoder of the transformer does not serve any functional purpose in this context. The reason for retaining it in the architecture, as observed in DETR codebase, is primarily for maintaining consistency with the standard transformer architectures.

5.3.5 Final Outputs

Using the Hungarian matching algorithm, we obtain the final outputs, including bounding box coordinates (x, y, w, h) with a shape of $num\_queries \times 4$, and classification scores with a shape of $num\_queries \times 1$.

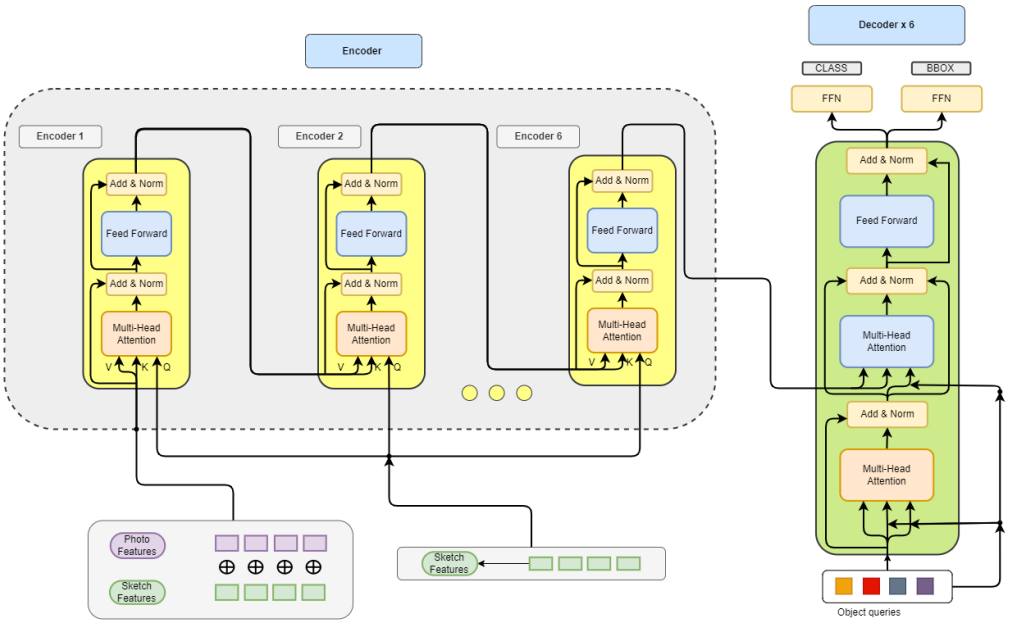

Figure 5.1: Sketch Canvas DETR Architecture

5.4 Exploring FG-SBIR Using Synthetic Sketches

In the initial stage of our research, we investigated synthetic sketches as a promising solution to address the data scarcity issues mentioned in Section 1.4.3. We conducted baseline experiments for the Fine-Grained Sketch-Based Image Retrieval (FG-SBIR) task using the ShoeV2 dataset. We explored three major sketch generation algorithms: Clipasso [136], Vinker et al. [137], and Chan et al. [18].

Considering factors such as ease of use and the time required to generate sketches, we opted to proceed with the approach presented by Caroline et al. [18]. We combined synthetic sketches with human-drawn sketches at various ratios and studied their impact on retrieval metrics. The results of this study are presented in Figure 5.2, 5.3.

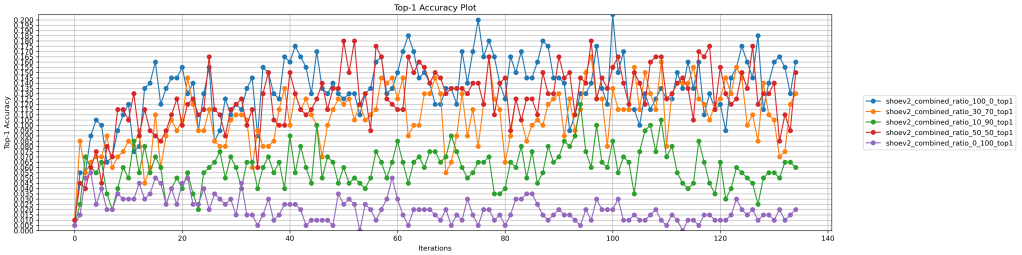

Figure 5.2: Top-1 Retrieval scores for FG-SBIR using various ratios of hand drawn and synthetic sketches

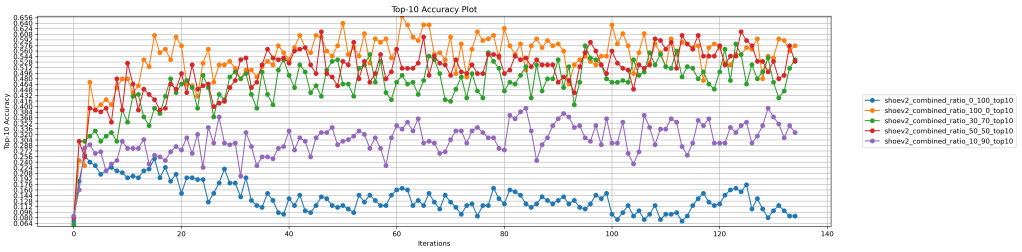

Figure 5.3: Top-10 Retrieval scores for FG-SBIR using various ratios of hand drawn and synthetic sketches

We conclude that using a ratio of 50% hand-drawn sketches and 50% synthetic sketches gave satisfactory results and reduces our dependency on collecting human drawn sketches for downstream sketch applications.

5.5 Dataset Preparation

In Sketch-Guided Object Localization (SGOL), data scarcity is a significant challenge for cross-modal learning algorithms. Acquiring paired photos and hand-drawn sketches is inherently difficult, limiting training data availability for SBIR and SGOL.

To address these challenges, we propose restructuring the Sketchy-COCO dataset. Our approach involves two key steps:

1. **Background Class Filtering:** We filter out background classes, focusing on foreground instances of interest.

2. **Single Instance Images:** Instead of using images with multiple instances, we create individual images, each featuring only one sketch instance (see Figure 5.4). This serves as both advanced data augmentation and improves model generalization.

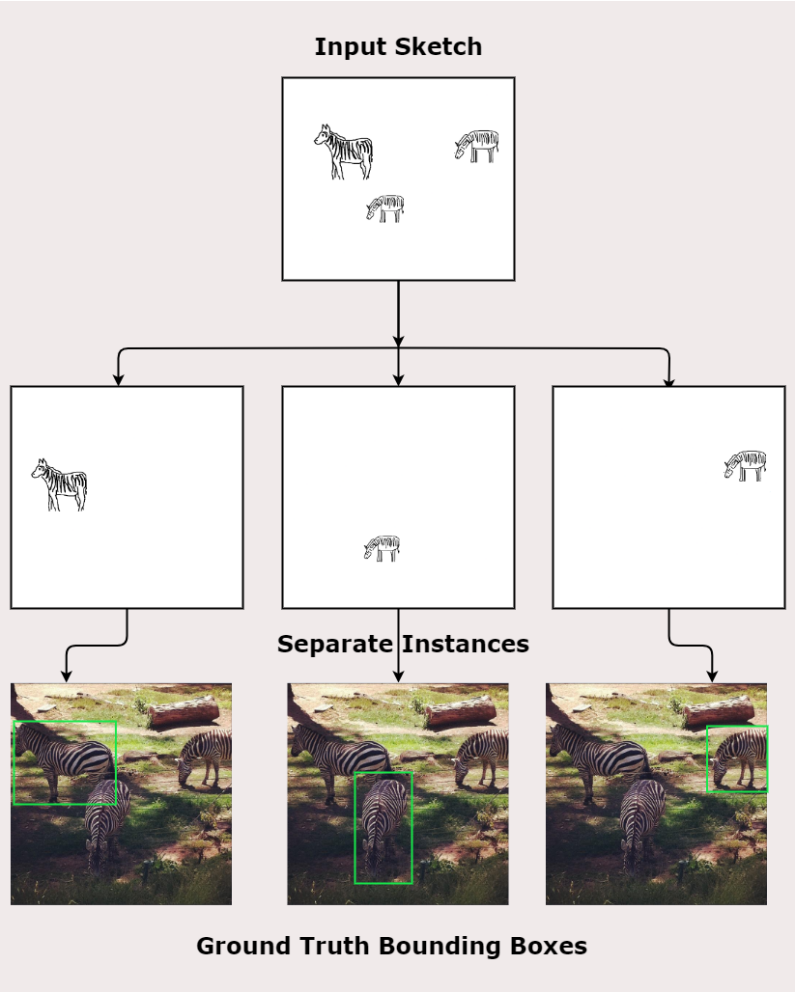

Figure 5.4: Single Instance Images with human drawn sketches

3. **Synthetic Sketches:** As described in Section 5.4 we explore using synthetic sketches (see Figure 5.5) in a 50:50 ratio with human-drawn sketches. However, this approach showed a slight drop in performance for the task of object detection.

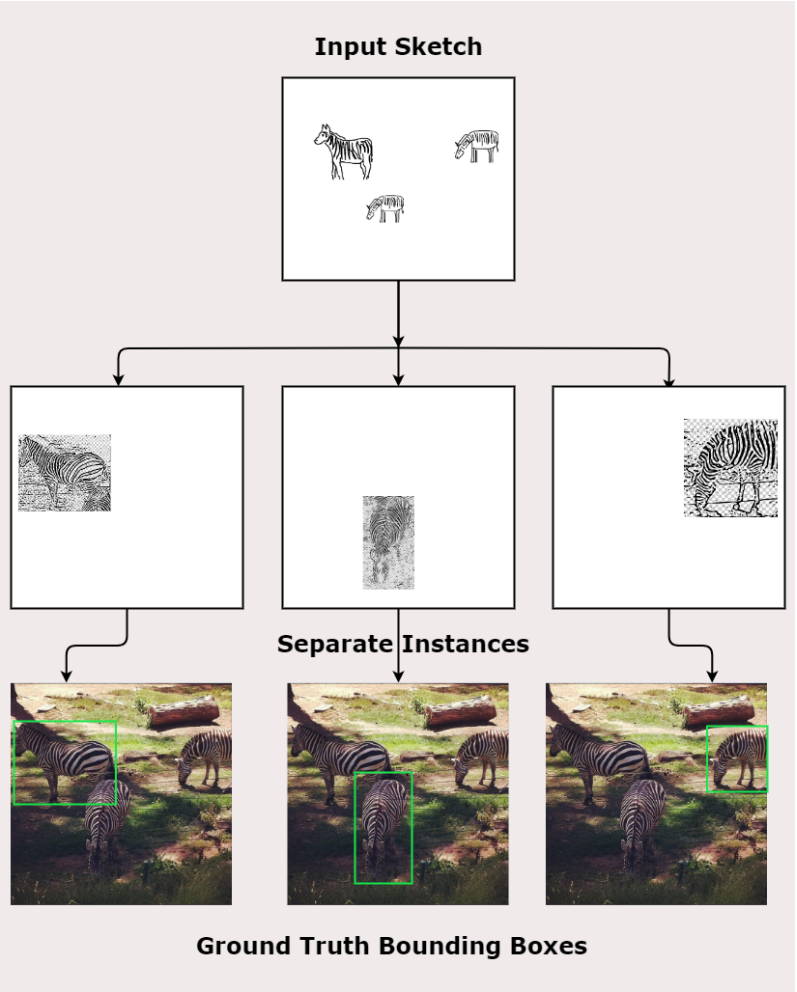

Figure 5.5: Single Instance Images with synthetic sketches

5.6 Multi-modal Data Ingestion and feature extraction

We follow standard normalization techniques to pre-process our sketch and image inputs and pass them through a feature extraction module, that can generate embeddings that can be ingested by the transformer encoder.

ResNet-50 [47] is a convolutional neural network architecture highly cited for image-related tasks. It contains residual blocks that mitigate the vanishing gradient problem, allowing for effective training of extremely deep networks.

Conventional approaches employ separate feature extractors for each modality. However, in our investigation we found that this leads to larger memory footprint.

In our work, we capitalize on ResNet-50 as a shared feature extractor for both sketch and image inputs. As a result, overall memory footprint of the final model is also reduced.

We take the output embeddings from the 4th layer of the ResNet architecture to feed our transformer encoder (see Figure 5.4). In our studies we discover that the suitable feature representations are obtained for both sketch and image using same backbone. The representations from different layers are visually presented in Figure 5.7.

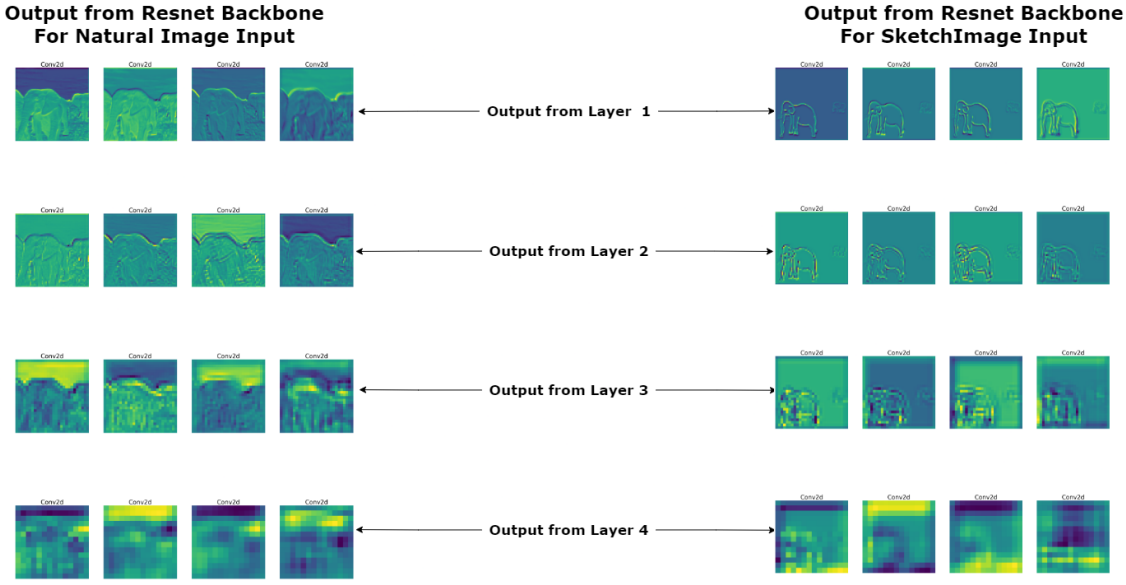

Figure 5.6: Outputs from intermediate layers of Resnet-50

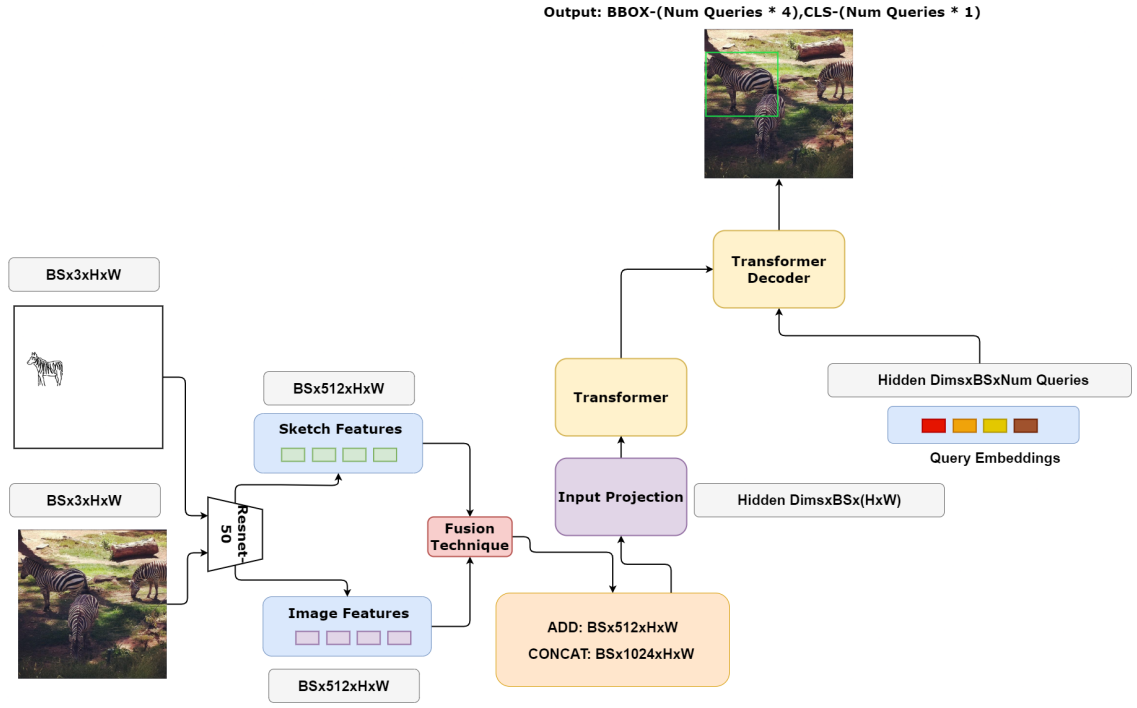

Figure 5.7: Data Ingestion pipeline and Data Flow architecture

5.7 Multi-modal Fusion Techniques

Once the input embeddings are generated, they need to be prepared or combined in a way that's suitable for the transformer module to ingest. The fusion process involves taking the individual embeddings of different input elements and combining them to create a unified representation that can be fed into the transformer. To combine the sketch and image modalities, we explore various fusion methods, which are discussed in Section 2.9. Initially, we attempted a straightforward approach by performing early element-wise addition of modalities. Surprisingly, this simple early fusion method yielded an $AP_{.5}$ score of 72.5%.

However, we questioned the validity of addition as a fusion method. Element-wise addition merges the elements, but in a transformer encoder layer, self-attention is crucial as it pays attention to individual tokens. In Element-wise addition, the final representation becomes a combined form. Therefore, we decided to experiment with early concatenation, where both modalities' content is preserved as Concatenation simply stacks the features thereby allowing the self-attention block to establish more meaningful connections.

Implementing and training early concatenation proved more challenging. The input projection block, responsible for projecting from the ResNet dimension to the transformer hidden dimension, could not be used due to change in layer dimensions. Consequently, we couldn't use pretrained weights from a DETR model trained on the COCO dataset. However, we manually adapted the code, keeping all weights from the DETR pre-trained model except for the input projection layer. This necessitated more training epochs, but the result was a significant increase in the $AP_{.5}$ score, reaching 82.1%.

5.7.1 Single Encoder Cross Attention Block

We explored other fusion methods like Hierarchical Attention, but they required multiple encoders with multiple layers, which introduced more parameters and computational limitations. As a workaround, we attempted to mimic a cross-attention-based block using only one encoder module. We fed image features to the key and values and sketch features to the query part of the transformer. In subsequent layers, we repeatedly used the initial sketch features as queries, while the consecutive hidden representations served as keys and values.

This approach improved accuracy overall. However, during qualitative assessment, we observed instances where it detected objects not present in the sketches. To address this issue, we sought a way to incorporate sketch features into the image features. We devised a simple trick based on our discussion of fusion techniques. We passed the element-wise addition of photo and sketch features as keys and values, while sketches served as queries. This yielded the best results both quantitatively and qualitatively, with an $AP_{.5}$ of 82.7% and no false detections.

As a result, we use this approach and integrate it into our final version of Sketch Canvas DETR as shown in Figure 5.1

5.8 Experiments

The chosen evaluation metric for our assessment is $AP_{0.5}$, as detailed in Section 2.13. From the results presented in Table 5.1, it is evident that the highest score was achieved when utilizing the cross-attention module described in Section 5.7.1, where the query consists of the sketch, and both key and value incorporate information from both the sketch and photo modalities.

The results reveal the impact of various fusion techniques. Simple fusion methods, such as addition, yielded satisfactory results, while more complex techniques like concatenation demon-

strated superior performance. Additionally, we observed that employing a shared backbone did not significantly compromise accuracy, leading to lighter models and faster training times.

Table 5.1: Sketch Canvas DETR Experiments

Experiment Name (Sketch:S, Photo:P)	$AP_{.5}$
S, P Self-attention Element-wise Addition Fusion (Separate backbone)	74.5
S, P Self-attention Element-wise Addition Fusion with 50 percent synthetic sketches (Separate backbone)	71.5
S, P Self-attention Element-wise Addition Fusion (Shared backbone)	75.8
S, P Self-attention Element-wise Addition Fusion (Learnable Embeddings) (Shared backbone)	Failed
S, P Self-attention Concatenation Fusion (Shared backbone)	82.1
S, P Self-attention Element-wise Addition Fusion with decoder conditioning (Shared backbone)	71.5
S, P Cross-attention Encoder (Query: S, Key, Value: S add P) (refer Section 5.7.1)	82.7
S, P Cross-attention Encoder (Query: S, Key, Value: S concat P) (refer Section 5.7.1)	Failed

5.9 Results

We present the results obtained from our model with the highest evaluation metric score, which is the Sketch Canvas DETR model employing the cross-attention block, as discussed in Section 5.7.1. Our results showcase the performance of the model in handling user sketch queries that contain both single and multiple instances. This demonstrates the robustness and versatility of our approach.

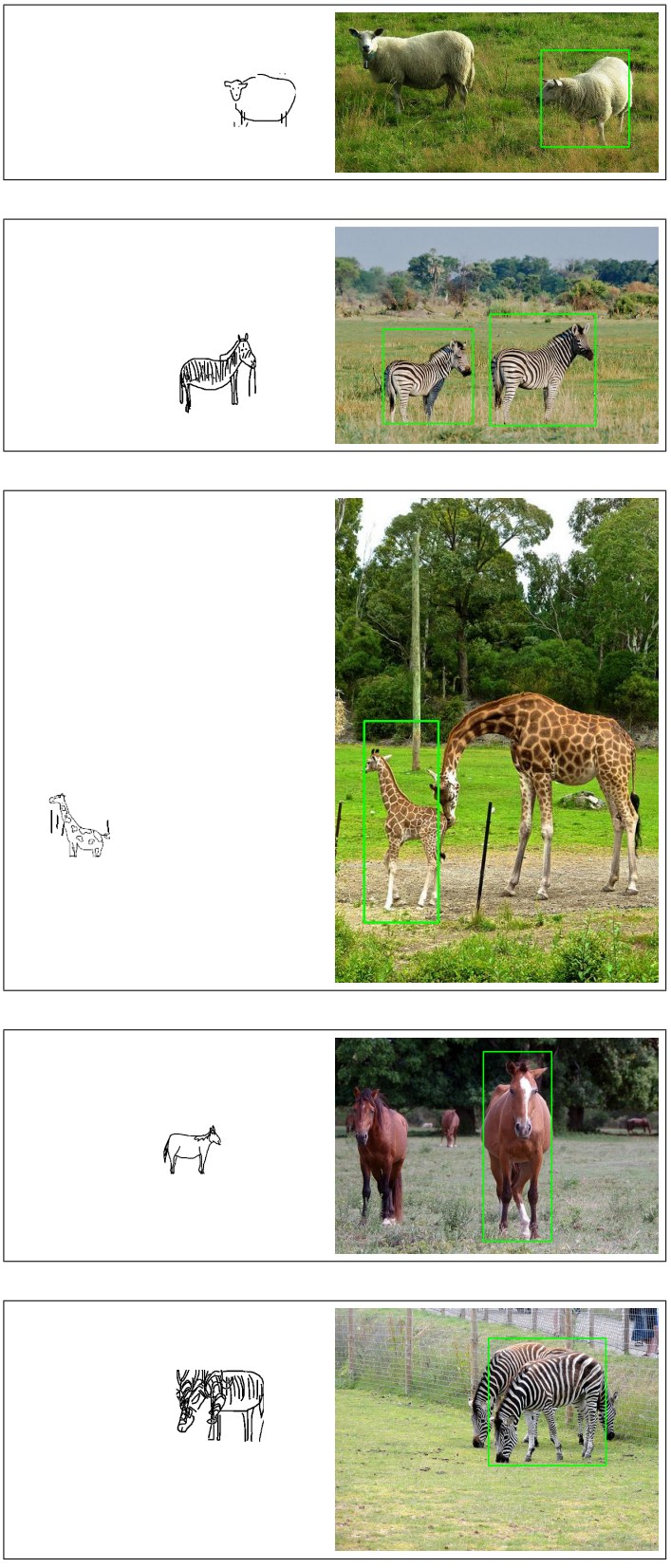

Figure 5.12: Single instance sketch queries: Results - Part 1

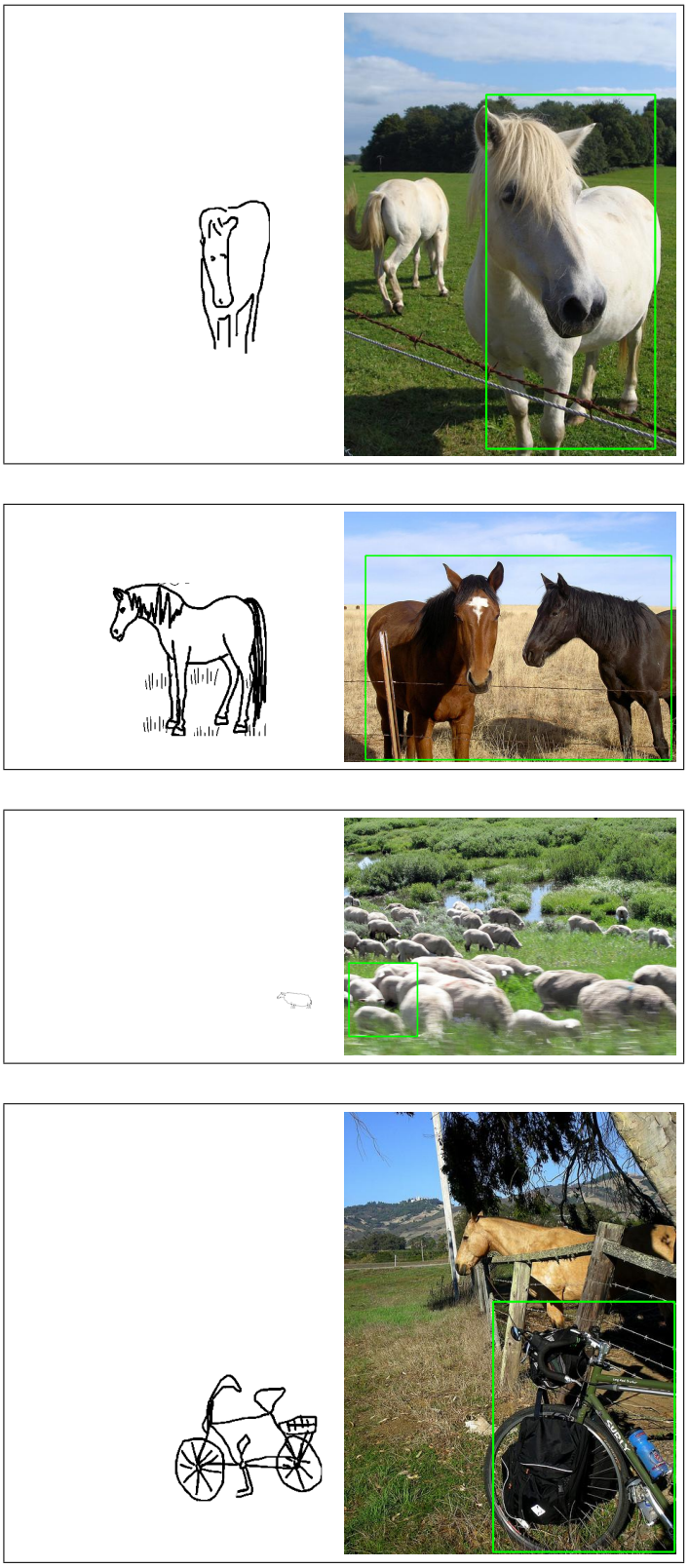

Figure 5.16: Single instance sketch queries: Results - Part 2

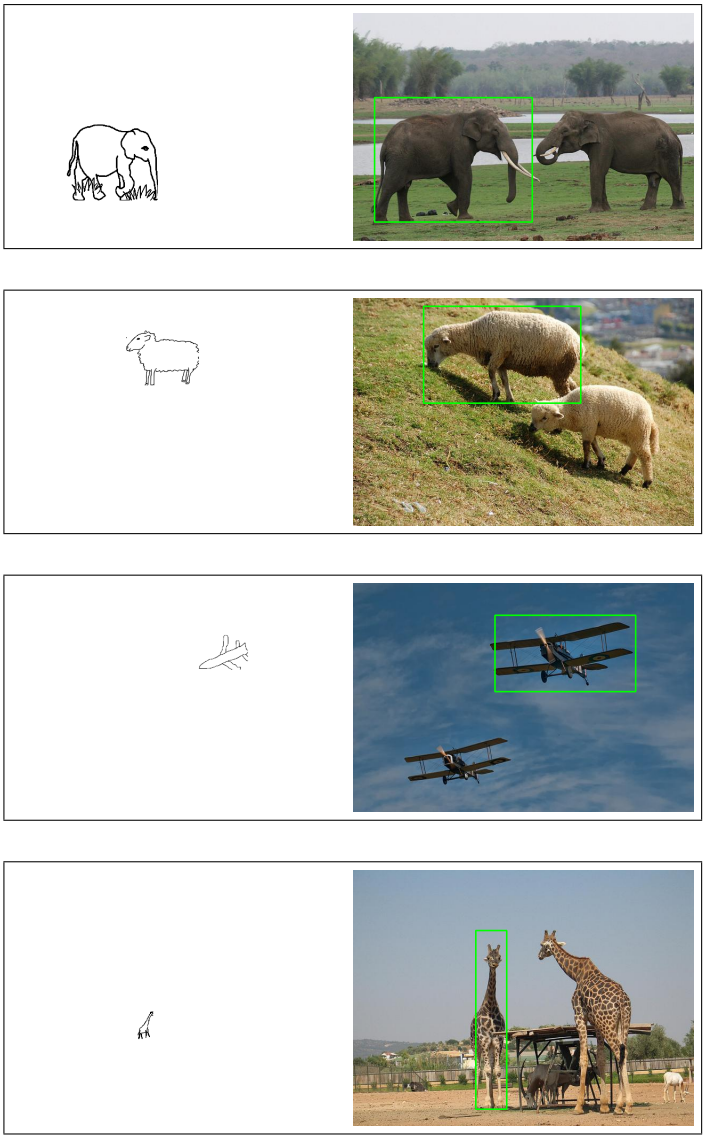

Figure 5.20: Single instance sketch queries: Results - Part 3

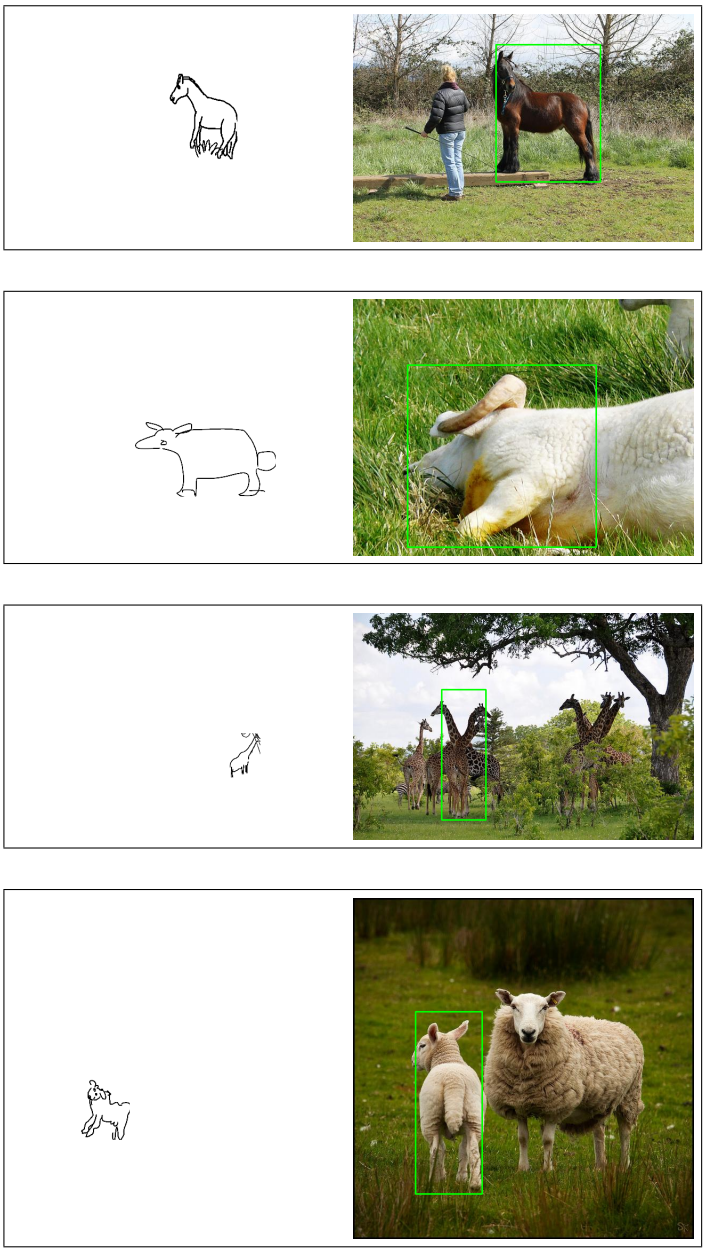

Figure 5.24: Single instance sketch queries: Results - Part 4

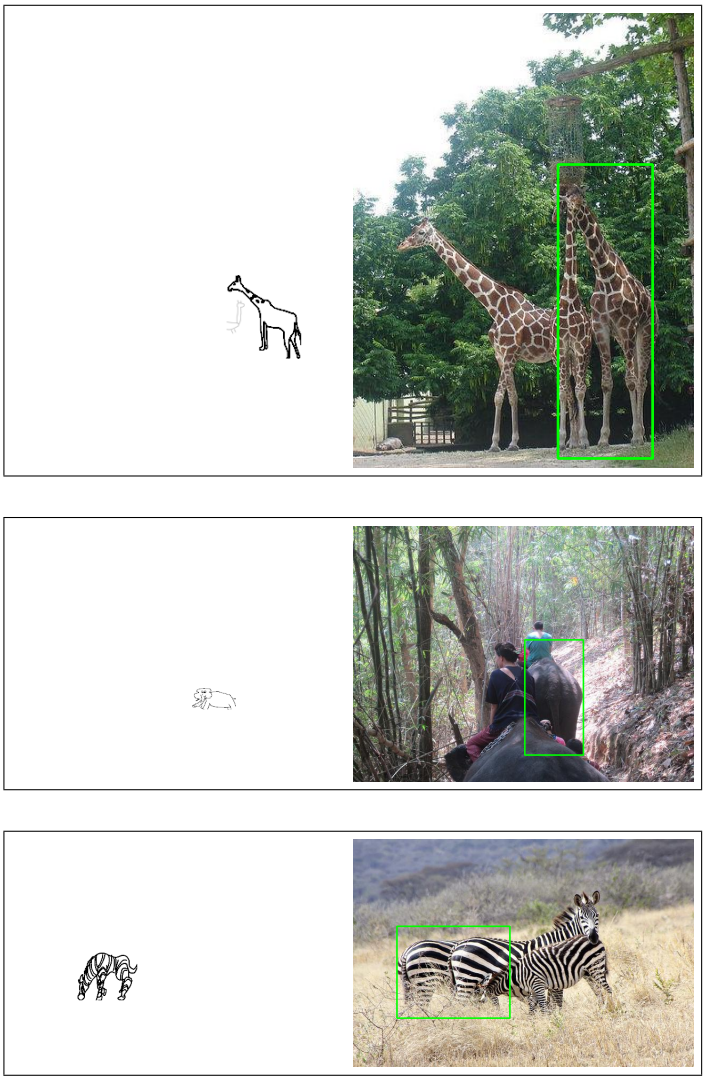

Figure 5.27: Single instance sketch queries: Results - Part 5

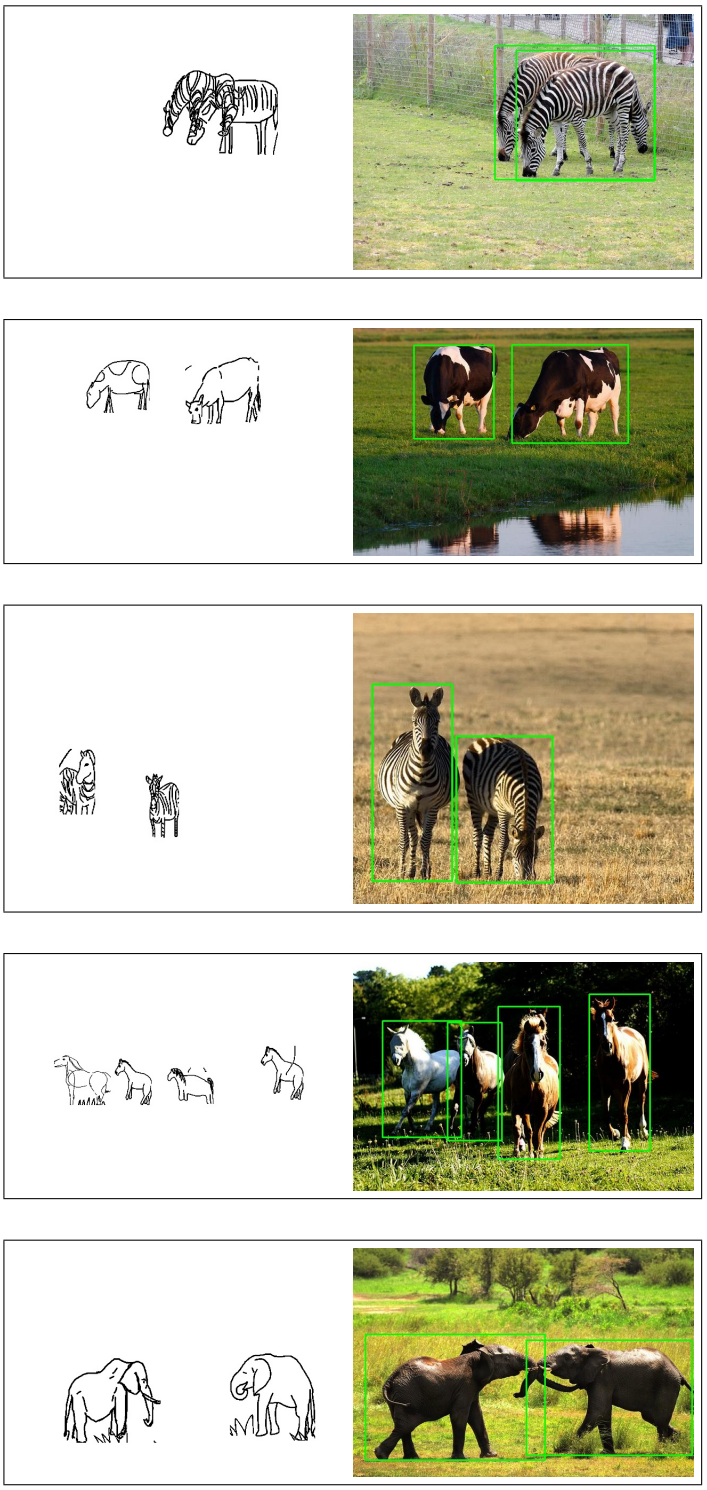

Figure 5.32: Multiple instance sketch queries: Results - Part 1

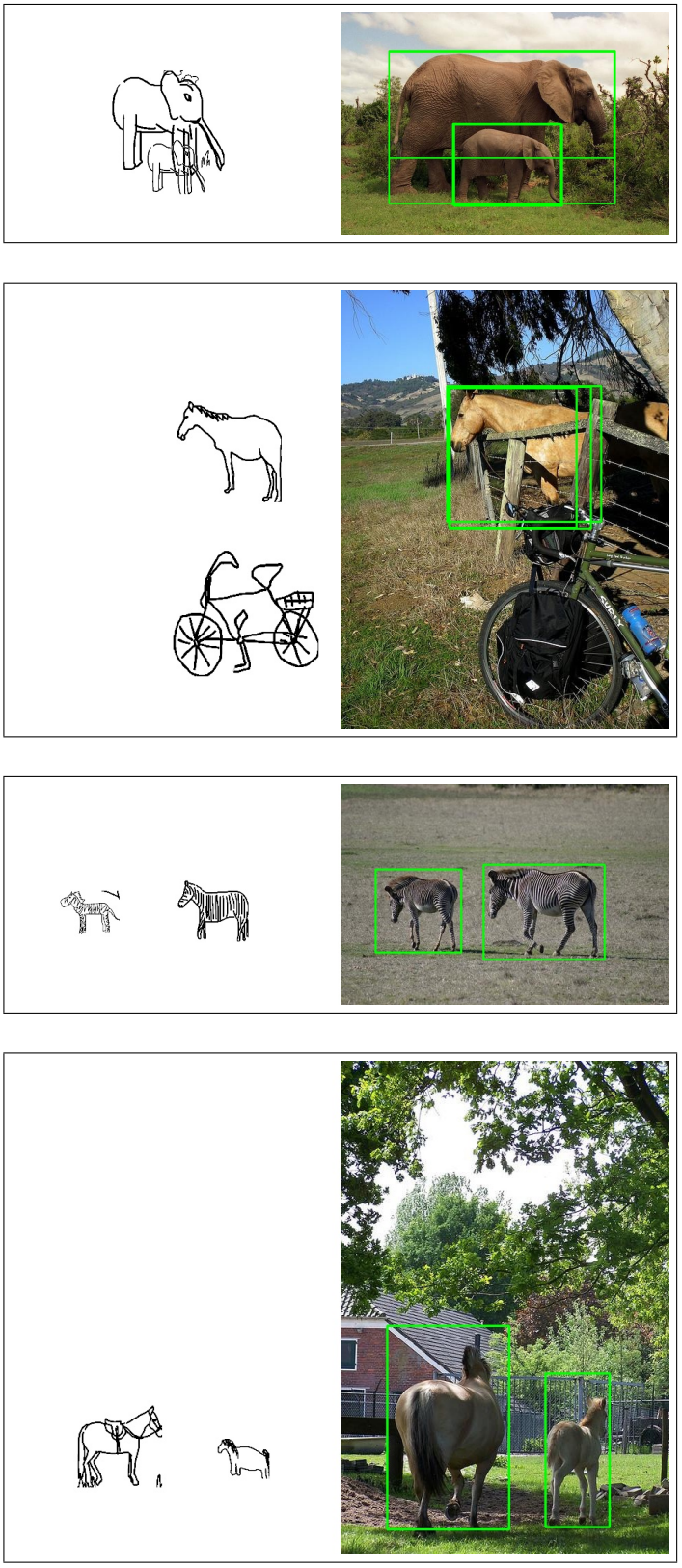

Figure 5.36: Multiple instance sketch queries:Results - Part 2

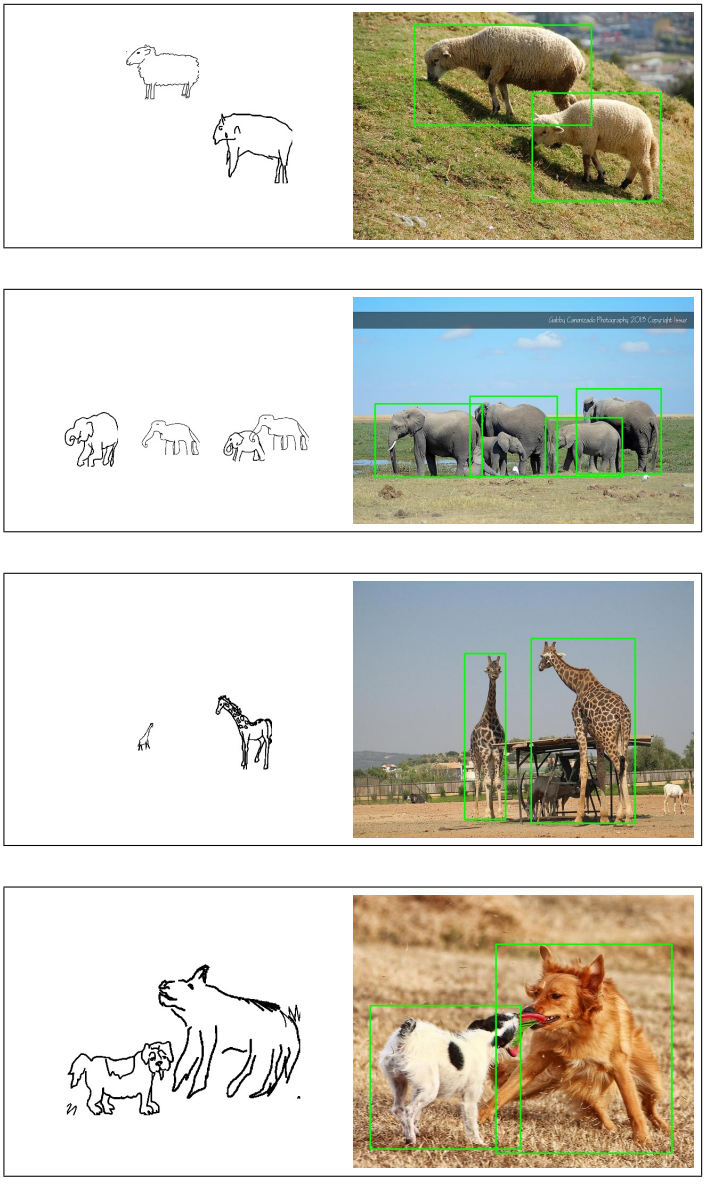

Figure 5.40: Multiple instance sketch queries:Results - Part 3

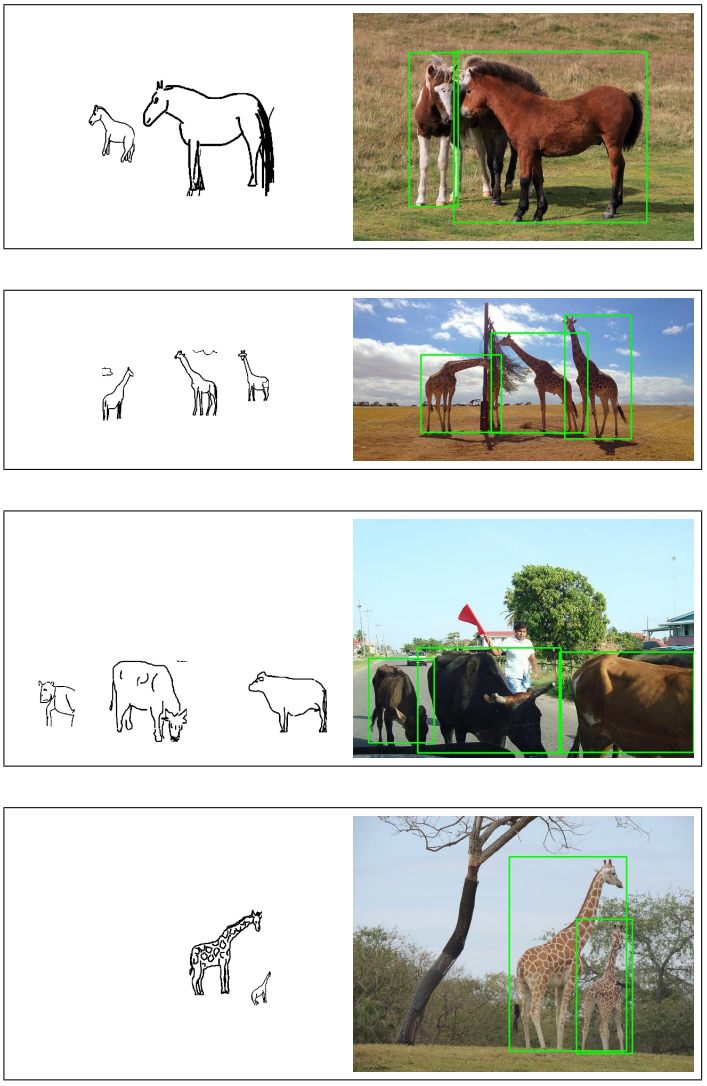

Figure 5.44: Multiple instance sketch queries:Results - Part 4

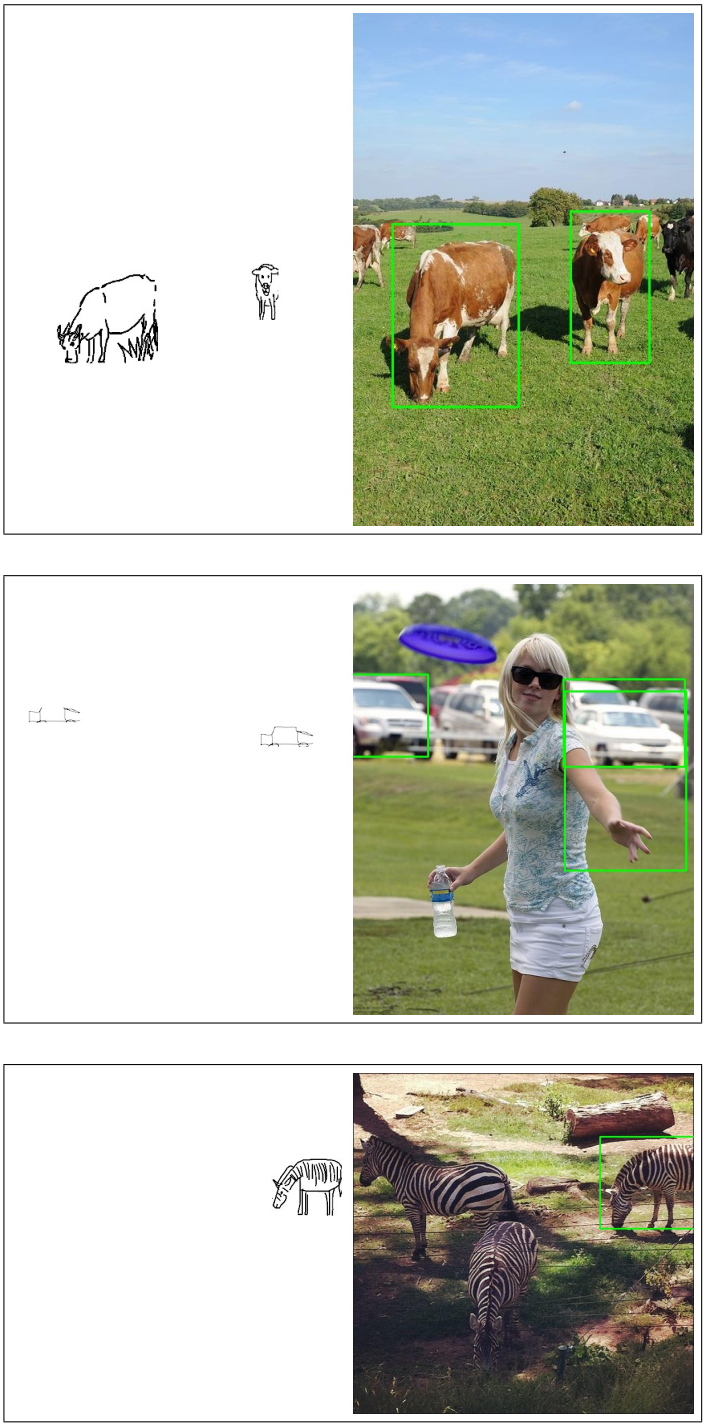

Figure 5.47: Multiple instance sketch queries:Results - Part 5

6 CONCLUSIONS

In the course of this thesis, we conducted baseline experiments alongside algorithms designed to detect objects in natural images using sketches as queries. We addressed specific limitations discussed in Section 1.4 by providing innovative solutions.

6.1 Future Work

Through our experiments, we identified a critical bottleneck in the current method, namely the ResNet feature extractor. We recognized that superior initial embeddings can significantly impact the performance of the transformer.

To overcome this challenge, we propose the following directions for future work:

- Utilize CLIP embeddings [94] to enhance initial embeddings. CLIP's models, pretrained on text, offer the potential for superior embeddings and can be invaluable in open vocabulary object detection scenarios.
- Focus on aligning sketch tokens closely with bounding box tokens within the natural image. This alignment can be achieved through the application of contrastive loss methods.
- Consider the possibility that sketches may not require a separate backbone. Exploring alternatives, such as representing the sketch as a vector graphic or line drawing, could improve overall performance. Since transformers excel at handling sequences, this approach could be particularly beneficial.

We are excited to announce the open-sourcing of the Sketch Canvas DETR codebase. We hope that this work will inspire and accelerate research in this field in the years to come.

BIBLIOGRAPHY

[1] Alaniz, S. , Mancini, M. , Dutta, A. , Marcos, D. , and Akata, Z. . Abstracting sketches through simple primitives. In *ECCV*, 2022.

[2] Bahng, H. , Jahanian, A. , Sankaranarayanan, S. , and Isola, P. . Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*, 2022.

[3] Bendale, A. and Boult, T. . Towards Open World Recognition. In *CVPR*, 2015.

[4] Bhunia, A. K. , Das, A. , Muhammad, U. R. , Yang, Y. , Hospedales, T. M. , Xiang, T. , Gryaditskaya, Y. , and Song, Y.-Z. . Pixelor: a competitive sketching ai agent. so you think you can sketch? *ACM TOG*, 2020.

[5] Bhunia, A. K. , Yang, Y. , Hospedales, T. M. , Xiang, T. , and Song, Y.-Z. . Sketch less for more: On-the-fly fine-grained sketch based image retrieval. Feb 2020. URL `http://arxiv.org/abs/2002.10310v4`.

[6] Bhunia, A. K. , Yang, Y. , Hospedales, T. M. , Xiang, T. , and Song, Y.-Z. . Sketch less for more: On-the-fly fine-grained sketch based image retrieval. In *CVPR*, 2020.

[7] Bhunia, A. K. , Chowdhury, P. N. , Sain, A. , and Song, Y.-Z. . Towards the unseen: Iterative text recognition by distilling from errors. In *ICCV*, 2021.

[8] Bhunia, A. K. , Chowdhury, P. N. , Sain, A. , and Song, Y.-Z. . Towards the unseen: Iterative text recognition by distilling from errors. In *ICCV*, 2021.

[9] Bhunia, A. K. , Chowdhury, P. N. , Sain, A. , Yang, Y. , Xiang, T. , and Song, Y.-Z. . More photos are all you need: Semi-supervised learning for fine-grained sketch based image retrieval. In *CVPR*, 2021.

[10] Bhunia, A. K. , Chowdhury, P. N. , Yang, Y. , Hospedales, T. M. , Xiang, T. , and Song, Y.-Z. . Vectorization and rasterization: Self-supervised learning for sketch and handwriting. In *CVPR*, 2021.

[11] Bhunia, A. K. , Gajjala, V. R. , Koley, S. , Kundu, R. , Sain, A. , Xiang, T. , and Song, Y.-Z. . Doodle it yourself: Class incremental learning by drawing a few sketches. In *CVPR*, 2022.

[12] Bhunia, A. K. , Koley, S. , Khilji, A. F. U. R. , Sain, A. , Chowdhury, P. N. , Xiang, T. , and Song, Y.-Z. . Sketching without worrying: Noise-tolerant sketch-based image retrieval. In *CVPR*, 2022.

[13] Bhunia, A. K. , Sain, A. , Shah, P. , Gupta, A. , Chowdhury, P. N. , Xiang, T. , and Song, Y.-Z. . Adaptive fine-grained sketch-based image retrieval. In *ECCV*, 2022.

[14] Bhunia, A. K. , Koley, S. , Kumar, A. , Sain, A. , Chowdhury, P. N. , Xiang, T. , and Song, Y.-Z. . Sketch2Saliency: Learning to Detect Salient Objects from Human Drawings. In *CVPR*, 2023.

[15] Bilen, H. and Vedaldi, A. . Weakly supervised deep detection networks. In *CVPR*, 2016.

[16] Carion, N. , Massa, F. , Synnaeve, G. , Usunier, N. , Kirillov, A. , and Zagoruyko, S. . End-to-end object detection with transformers. In *ECCV*, 2020.

[17] Caron, M. , Touvron, H. , Misra, I. , Jegou, H. , Mairal, J. , Bojanowski, P. , and Joulin, A. . Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

[18] Chan, C. , Durand, F. , and Isola, P. . Learning to generate line drawings that convey geometry and semantics, 2022.

[19] Chen, P. , Liu, S. , Zhao, H. , and Jia, J. . Gridmask data augmentation. *arXiv preprint arXiv:2001.04086*, 2020.

[20] Chowdhury, P. N. , Bhunia, A. K. , Gajjala, V. R. , Sain, A. , Xiang, T. , and Song, Y.-Z. . Partially Does It: towards scene-level FG-SBIR with partial input. In *CVPR*, 2022.

[21] Chowdhury, P. N. , Sain, A. , Gryaditskaya, Y. , Bhunia, A. K. , Xiang, T. , and Song, Y.-Z. . Fs-coco: Towards understanding of freehand sketches of common objects in context. In *ECCV*, 2022.

[22] Chowdhury, P. N. , Wang, T. , Ceylan, D. , Song, Y.-Z. , and Gryaditskaya, Y. . Garment ideation: Iterative view-aware sketch-based garment modeling. In *3DV*, 2022.

[23] Chowdhury, P. N. , Bhunia, A. K. , Sain, A. , Koley, S. , Xiang, T. , and Song, Y.-Z. . SceneTrilogy: On Human Scene-Sketch and its Complementarity with Photo and Text. In *CVPR*, 2023.

[24] Chowdhury, P. N. , Bhunia, A. K. , Sain, A. , Koley, S. , Xiang, T. , and Song, Y.-Z. . What can human sketches do for object detection? Mar 2023. URL http://arxiv.org/abs/2303.15149v1.

[25] Collomosse, J. , Bui, T. , and Hailin, J. . Livesketch: Query perturbations for guided sketch-based visual search. In *CVPR*, 2019.

[26] Das, A. , Yang, Y. , Hospedales, T. , Xiang, T. , and Song, Y.-Z. . Sketchode: Learning neural sketch representation in continuous time. In *ICLR*, 2022.

[27] Devlin, J. , Chang, M.-W. , Lee, K. , and Toutanova, K. . Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[28] DeVries, T. and Taylor, G. W. . Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[29] Dey, S. , Riba, P. , Dutta, A. , Llados, J. , and Song, Y.-Z. . Doodle to search: Practical zero-shot sketch-based image retrieval. In *CVPR*, 2019.

[30] Dey, S. , Riba, P. , Dutta, A. , Llados, J. , and Song, Y.-Z. . Doodle to search: Practical zero-shot sketch-based image retrieval. Apr 2019. doi: 10.1109/CVPR.2019.00228. URL http://arxiv.org/abs/1904.03451v2. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

[31] Dey, S. , Riba, P. , Dutta, A. , Llados, J. L. , and Song, Y.-Z. . Doodle to search: Practical zero-shot sketch-based image retrieval. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2019. doi: 10.1109/cvpr.2019.00228. URL https://doi.org/10.1109%2Fcvpr.2019.00228.

[32] Diba, A. , Sharma, V. , Pazandeh, A. , Pirsiavash, H. , and Gool, L. V. . Weakly supervised cascaded convolutional networks. In *CVPR*, 2017.

[33] Dietterich, T. G. , Lathrop, R. H. , and Lozano-Pérez, T. . Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 1997.

[34] Dietterich, T. G. , Lathrop, R. H. , and Lozano-Pérez, T. . Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 1997.

[35] Dosovitskiy, A. , Beyer, L. , Kolesnikov, A. , Weissenborn, D. , Zhai, X. , Unterthiner, T. , Dehghani, M. , Minderer, M. , Heigold, G. , Gelly, S. , Uszkoreit, J. , and Houlsby, N. . An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020.

[36] Everingham, M. , Gool, L. V. , Williams, C. K. I. , Winn, J. , and Zisserman, A. . The pascal visual object classes (voc) challenge. *IJCV*, 2010.

[37] Ganin, Y. and Lempitsky, V. . Unsupervised domain adaptation by backpropagation. In *ICML*, 2015.

[38] Gao, C. , Liu, Q. , Xu, Q. , Wang, L. , Liu, J. , and Zou, C. . Sketchycoco: Image generation from freehand scene sketches, 2020.

[39] Gao, C. , Liu, Q. , Xu, Q. , Wang, L. , Liu, J. , and Zou, C. . Sketchycoco: Image generation from freehand scene sketches. In *CVPR*, 2020.

[40] Girshick, R. . Fast-rcnn. In *ICCV*, 2015.

[41] Girshick, R. , Donahue, J. , Darrell, T. , and Malik, J. . Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[42] Goodwin, T. , Vollick, I. , and Hertzmann, A. . Isophote distance: A shading approach to artistic stroke thickness. In *NPAR*, 2007.

[43] Gu, X. , Lin, T.-Y. , Kuo, W. , and Cui, Y. . Open-vocabulary object detection via vision and language knowledge distillation. In *ICLR*, 2022.

[44] Ha, D. and Eck, D. . A neural representation of sketch drawings. In *ICLR*, 2018.

[45] Ham, C. , Tarres, G. C. , Bui, T. , Hays, J. , Lin, Z. , and Collomosse, J. . Cogs: Controllable generation and search from sketch and style. In *ECCV*, 2022.

[46] He, K. , Zhang, X. , Ren, S. , and Sun, J. . Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*, 2014.

[47] He, K. , Zhang, X. , Ren, S. , and Sun, J. . Deep residual learning for image recognition. In *CVPR*, 2016.

[48] He, K. , Gkioxari, G. , Dollár, P. , and Girshick, R. . Mask r-cnn. In *ICCV*, 2017.

[49] Hendricks, L. A. , Burns, K. , Saenko, K. , Darrel, T. , and Rohrbach, A. . Women also snowboard: Overcoming bias in captioning models. In *ECCV*, 2018.

[50] Heo, B. , Kim, J. , Yun, S. , Park, H. , Kwak, N. , and Choi, J. Y. . A comprehensive overhaul of feature distillation. In *ICCV*, 2019.

[51] Hertzmann, A. . Why do line drawings work? a realism hypothesis. *Perception*, 2020.

[52] Horiguchi, S. , Ikami, D. , and Aizawa, K. . Significance of softmax-based features in comparison to distance metric learning-based features. *IEEE-TPAMI*, 2019.

[53] Hsieh, T.-I. , Lo, Y.-C. , Chen, H.-T. , and Liu, T.-L. . One-shot object detection with co-attention and co-excitation. In *NeurIPS*, 2019.

[54] Hu, C. , Li, D. , Song, Y.-Z. , Xiang, T. , and Hospedales, T. M. . Sketch-a-classifier: Sketch-based photo classifier generation. In *CVPR*, 2018.

[55] Hu, C. , Li, D. , Yang, Y. , Hospedales, T. M. , and Song, Y.-Z. . Sketch-a-segmenter: Sketch-based photo segmenter generation. *IEEE-TIP*, 2020.

[56] Huang, G. , Laradji, I. , Vazquez, D. , Lacoste-Julien, S. , and Rodriguez, P. . A survey of self-supervised and few-shot object detection. *IEEE-PAMI*, 2022.

[57] Huang, Z. , Zou, Y. , Bhagavatula, V. , and Huang, D. . Comprehensive attention self-distillation for weakly-supervised object detection. In *NeurIPS*, 2020.

[58] Jia, M. , Tang, L. , Chen, B.-C. , Cardie, C. , Belongie, S. , Hariharan, B. , and Lim, S.-N. . Visual prompt tuning. In *ECCV*, 2022.

[59] Jie, Z. , Wei, Y. , Jin, X. , Feng, J. , and Liu, W. . Deep self-taught learning for weakly supervised object localization. In *CVPR*, 2017.

[60] Joseph, K. J. , Khan, S. , Khan, F. S. , and Balasubramanian, V. N. . Towards Open World Object Detection. In *CVPR*, 2021.

[61] Kamath, A. , Singh, M. , LeCun, Y. , Synnaeve, G. , Misra, I. , and Carion, N. . Mdetr – modulated detection for end-to-end multi-modal understanding. Apr 2021. URL `http://arxiv.org/abs/2104.12763v2`.

[62] Kennedy, J. M. . A psychology of picture perception: Images and information. *Jossey-Bass Publishers*, 1974.

[63] Khattak, M. U. , Rasheed, H. , Maaz, M. , Khan, S. , and Khan, F. S. . Maple: Multi-modal prompt learning. *arXiv preprint arXiv:2210.03117*, 2022.

[64] Kobayashi, K. , Gu, L. , Hataya, R. , Mizuno, T. , Miyake, M. , Watanabe, H. , Takahashi, M. , Takamizawa, Y. , Yoshida, Y. , Nakamura, S. , Kouno, N. , Bolatkan, A. , Kurose, Y. , Harada, T. , and Hamamoto, R. . Sketch-based Medical Image Retrieval. *arXiv preprint arXiv:2303.03633*, 2023.

[65] Koley, S. , Bhunia, A. K. , Sain, A. , Chowdhury, P. N. , Xiang, T. , and Song, Y.-Z. . Picture that Sketch: Photorealistic Image Generation from Abstract Sketches. In *CVPR*, 2023.

[66] Krishna, R. , Zhu, Y. , Groth, O. , Johnson, J. , Hata, K. , Kravitz, J. , Chen, S. , Kalantidis, Y. , Li, L.-J. , Shamma, D. A. , Bernstein, M. , and Fei-Fei, L. . Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2017.

[67] Krizhevsky, A. , Sutskever, I. , and Hinton, G. E. . Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

[68] Law, H. and Deng, J. . Cornernet: Detecting objects as paired keypoints. In *ECCV*, 2018.

[69] Lee, C. , Park, S. , Song, H. , Ryu, J. , Kim, S. , Kim, H. , Pereira, S. , and Yoo, D. . Interactive multi-class tiny-object detection. In *CVPR*, 2022.

[70] Li, C. , Yang, J. , Zhang, P. , Gao, M. , Xiao, B. , Dai, X. , Yuan, L. , and Gao, J. . Efficient self-supervised vision transformers for representation learning. In *ICLR*, 2022.

[71] Li, D. , Huang, J.-B. , Li, Y. , Wang, S. , and Yang, M.-H. . Weakly supervised object localization with progressive domain adaptation. In *CVPR*, 2016.

[72] Li, G. , Jampani, V. , Sevilla-Lara, L. , Sun, D. , Kim, J. , and Kim, J. . Adaptive prototype learning and allocation for few-shot segmentation. In *CVPR*, 2021.

[73] Li, P. , Li, X. , and Long, X. . Fencemask: A data augmentation approach for pre-extracted image features. *arXiv preprint arXiv:2006.07877*, 2020.

[74] Liang, F. , Wu, B. , Dai, X. , Li, K. , Zhao, Y. , Zhang, H. , Zhang, P. , Vajda, P. , and Marculescu, D. . Open-vocabulary semantic segmentation with mask-adapted clip. *arXiv preprint arXiv:2210.04150*, 2022.

[75] Lin, T.-Y. , Maire, M. , Belongie, S. , Hays, J. , Perona, P. , Ramanan, D. , Dollár, P. , and Zitnick, C. L. . Microsoft coco: Common objects in context. In *ECCV*, 2014.

[76] Lin, T.-Y. , , Dollár, P. , , Girshick, R. , He, K. , Hariharan, B. , and Belongie, S. . Feature pyramid networks for object detection. In *CVPR*, 2017.

[77] Lin, T.-Y. , Goyal, P. , Girshick, R. , He, K. , and Dollár, P. . Focal loss for dense object detection. In *ICCV*, 2017.

[78] Liu, F. , Zhou, C. , Deng, X. , Zuo, R. , Lai, Y.-K. , Ma, C. , Liu, Y.-J. , and Wang, H. . Scenesketcher: Fine-grained image retrieval with scene sketches. In *ECCV*, 2020.

[79] Liu, F.-L. , Chen, S.-Y. , Lai, Y.-K. , Li, C. , Jiang, Y.-R. , Fu, H. , and Gao, L. . DeepFaceVideoEditing: Sketch-based deep editing of face videos. *ACM TOG*, 2022.

[80] Liu, K. , Li, Y. , Xu, N. , and Natarajan, P. . Learn to combine modalities in multimodal deep learning. May 2018. URL `http://arxiv.org/abs/1805.11730v1`.

[81] Liu, L. , Shen, F. , Shen, Y. , Liu, X. , and Shao, L. . Deep sketch hashing: Fast free-hand sketch-based image retrieval. In *CVPR*, 2017.

[82] Liu, W. , Anguelov, D. , Erhan, D. , Szegedy, C. , Reed, S. , Fu, C.-Y. , and Berg, A. C. . Ssd: Single shot multibox detector. In *ECCV*, 2016.

[83] Ma, M. , Ren, J. , Zhao, L. , Testuggine, D. , and Peng, X. . Are multimodal transformers robust to missing modality?, 2022.

[84] Mikolov, T. , Chen, K. , Corrado, G. , and Dean, J. . Efficient estimation of word representations in vector space. In *ICLR*, 2013.

[85] Minderer, M. , Gritsenko, A. , Stone, A. , Neumann, M. , Weissenborn, D. , Dosovitskiy, A. , Mahendran, A. , Arnab, A. , Dehghani, M. , Shen, Z. , Wang, X. , Zhai, X. , Kipf, T. , and Houlsby, N. . Simple open-vocabulary object detection with vision transformers. In *ECCV*, 2022.

[86] Mouzenidis, P. , Louros, A. , Konstantinidis, D. , Dimitropoulos, K. , and Daras, P. . Multimodal variational faster-rcnn for improved visual object detection in manufacturing. In *ICCV*, 2021.

[87] Muhammad, U. R. , Yang, Y. , Song, Y.-Z. , Xiang, T. , and Hospedales, T. M. . Learning deep sketch abstraction. In *CVPR*, 2018.

[88] Pang, K. , Song, Y.-Z. , Xiang, T. , and Hospedales, T. M. . Cross-domain generative learning for fine-grained sketch-based image retrieval. In *BMVC*, 2017.

[89] Pang, K. , Li, K. , Yang, Y. , Zhang, H. , Hospedales, T. M. , Xiang, T. , and Song, Y.-Z. . Generalising fine-grained sketch-based image retrieval. In *CVPR*, 2019.

[90] Pang, K. , Yang, Y. , Hospedales, T. M. , Xiang, T. , and Song, Y.-Z. . Solving mixed-modal jigsaw puzzle for fine-grained sketch-based image retrieval. In *CVPR*, 2020.

[91] Pawłowski, M. , Wróblewska, A. , and Sysko-Romańczuk, S. . Effective techniques for multimodal data fusion: A comparative analysis. *Sensors*, 23(5), 2023. ISSN 1424-8220. doi: 10.3390/s23052381. URL https://www.mdpi.com/1424-8220/23/5/2381.

[92] Qi, A. , Gryaditskaya, Y. , Xiang, T. , and Song, Y.-Z. . One sketch for all: One-shot personalized sketch segmentation. *IEEE-TIP*, 2022.

[93] Radford, A. , Wu, J. , Child, R. , Luan, D. , Amodei, D. , and Sutskever, I. . Language models are unsupervised multitask learners. *OpenAI Blog*, 2019.

[94] Radford, A. , Kim, J. W. , Hallacy, C. , Ramesh, A. , Goh, G. , Agarwal, S. , Sastry, G. , Askell, A. , Mishkin, P. , Clark, J. , Krueger, G. , and Sutskever, I. . Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[95] Raffel, C. , Shazeer, N. , Roberts, A. , Lee, K. , Narang, S. , Matena, M. , Zhou, Y. , Li, W. , and Liu, P. J. . Exploring the limits of transfer learning with a unified text-to-text transformer. Oct 2019. URL http://arxiv.org/abs/1910.10683v3.

[96] Redmon, J. and Farhadi, A. . Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[97] Redmon, J. , Divvala, S. , Girshick, R. , and Farhadi, A. . You only look once: Unified, real-time object detection. In *CVPR*, 2016.

[98] Reed, W. J. . The pareto, zipf and other power laws. *Economic Letters*, 2001.

[99] Ren, S. , He, K. , Girshick, R. , and Sun, J. . Faster r-cnn: Towards real-time object detection. In *NeurIPS*, 2015.

[100] Ren, Z. , Yu, Z. , Yang, X. , Liu, M.-Y. , Lee, Y. J. , Schwing, Alexander, G. , and Kautz, J. . Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *CVPR*, 2020.

[101] Rezatofighi, H. , Tsoi, N. , Gwak, J. , Sadeghian, A. , Reid, I. , and Savarese, S. . Generalized intersection over union: A metric and a loss for bounding box regression. Feb 2019. URL `http://arxiv.org/abs/1902.09630v2`.

[102] Riba, P. , Dey, S. , Biten, A. F. , and Llados, J. . Localizing infinity-shaped fishes: Sketch-guided object localization in the wild. Sep 2021. URL `http://arxiv.org/abs/2109.11874v1`.

[103] Riba, P. , Dey, S. , Biten, A. F. , and Llados, J. . Localizing infinity-shaped fishes: Sketch-guided object localization in the wild. *arXiv preprint arXiv:2109.11874*, 2021.

[104] Ribeiro, L. S. F. , Bui, T. , Collomosse, J. , and Ponti, M. . Sketchformer: Transformer-based representation for sketched structure. In *CVPR*, 2020.

[105] Rombach, R. , Blattmann, A. , Lorenz, D. , Esser, P. , and Ommer, B. . High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[106] Sain, A. , Bhunia, A. K. , Yang, Y. , Xiang, T. , and Song, Y.-Z. . Stylemeup: Towards style-agnostic sketch-based image retrieval. In *CVPR*, 2021.

[107] Sain, A. , Bhunia, A. K. , Potlapalli, V. , Chowdhury, P. N. , Xiang, T. , and Song, Y.-Z. . Sketch3t: Test-time training for zero-shot sbir. In *CVPR*, 2022.

[108] Sain, A. , Bhunia, A. K. , Chowdhury, P. N. , Sain, A. , Koley, S. , Xiang, T. , and Song, Y.-Z. . CLIP for All Things Zero-Shot Sketch-Based Image Retrieval, Fine-Grained or Not. In *CVPR*, 2023.

[109] Sain, A. , Bhunia, A. K. , Koley, S. , Chowdhury, P. N. , Chattopadhyay, S. , Xiang, T. , and Song, Y.-Z. . Exploiting Unlabelled Photos for Stronger Fine-Grained SBIR. In *CVPR*, 2023.

[110] Sandler, M. , Zhmoginov, A. , Vladymyrov, M. , and Jackson, A. . Fine-tuning image transformers using learnable memory. In *CVPR*, 2022.

[111] Sangkloy, P. , Burnell, N. , Ham, C. , and Hays, J. . The sketchy database: Learning to retrieve badly drawn bunnies. *ACM Trans. Graph.*, 35(4), jul 2016. ISSN 0730-0301. doi: 10.1145/2897824.2925954. URL https://doi.org/10.1145/2897824.2925954.

[112] Sangkloy, P. , Burnell, N. , Ham, C. , and Hays, J. . The sketchy database: Learning to retrieve badly drawn bunnies. *ACM TOG*, 2016.

[113] Sayim, B. and Cavanagh, P. . What line drawings reveal about the visual brain. *Front. Hum. Neurosci.*, 2011.

[114] Shankar, S. , Piratla, V. , Chakrabarti, S. , Chaudhuri, S. , Jyothi, P. , and Sarawagi, S. . Generalizing across domains via cross-gradient training. In *ICLR*, 2018.

[115] Shao, F. , Chen, L. , Shao, J. , Ji, W. , Xiao, S. , Ye, L. , Zhuang, Y. , and Xiao, J. . Deep learning for weakly-supervised object detection and object localization: A survey, 2021.

[116] Shao, F. , Chen, L. , Shao, J. , Ji, W. , Xiao, S. , Ye, L. , Zhuang, Y. , and Xiao, J. . Deep learning for weakly-supervised object detection and localization: A survey. *Neurocomputing*, 2022.

[117] Shen, Y. , Liu, L. , Shen, F. , and Shao, L. . Zero-shot sketch-image hashing. In *CVPR*, 2018.

[118] Shen, Y. , Ji, R. , Wang, Y. , Wu, Y. , and Cao, L. . Cyclic guidance for weakly supervised joint detection and segmentation. In *CVPR*, 2019.

[119] Simonyan, K. and Zisserman, A. . Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[120] Singh, K. K. and Lee, Y. J. . Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *ICCV*, 2017.

[121] Song, J. , Song, Y.-Z. , Xiang, T. , and Hospedales, T. . Fine-grained image retrieval: the text/sketch input dilemma. In *BMVC*, 2017.

[122] Song, J. , Yu, Q. , Song, Y.-Z. , Xiang, T. , and Hospedales, T. M. . Deep spatial-semantic attention for fine-grained sketch-based image retrieval. In *ICCV*, 2017.

[123] Sun, P. , Jiang, Y. , Xie, E. , Shao, W. , Yuan, Z. , Wang, C. , and Luo, P. . What makes for end-to-end object detection? Dec 2020. URL http://arxiv.org/abs/2012.05780v2.

[124] Szegedy, C. , Liu, W. , Jia, Y. , Sermanet, P. , Reed, S. , Anguelov, D. , Erhan, D. , Vanhoucke, V. , and Rabinovich, A. . Going deeper with convolutions. In *CVPR*, 2015.

[125] Tang, P. , Wang, X. , Bai, X. , and Liu, W. . Multiple instance detection network with online instance classifier refinement. In *CVPR*, 2017.

[126] Tang, P. , Wang, X. , Bai, S. , Shen, W. , Bai, X. , Liu, W. , and Yuille, A. . Pcl: Proposal cluster learning for weakly supervised object detection. *IEEE-TPAMI*, 2018.

[127] Tang, P. , Wang, X. , Wang, A. , Yan, Y. , Liu, W. , Huang, J. , and Yuille, A. . Weakly supervised region proposal network and object detection. In *ECCV*, 2018.

[128] Torralba, A. and Efros, Alexei, A. . Unbiased look at dataset bias. In *CVPR*, 2011.

[129] Tripathi, A. , Dani, R. R. , Mishra, A. , and Chakraborty, A. . Sketch-guided object localization in natural images. In *ECCV*, 2020.

[130] Tripathi, A. , Dani, R. R. , Mishra, A. , and Chakraborty, A. . Sketch-guided object localization in natural images, 2020.

[131] Tripathi, A. , Mishra, A. , and Chakraborty, A. . Query-guided attention in vision transformers for localizing objects using a single sketch. Mar 2023. URL `http://arxiv.org/abs/2303.08784v1`.

[132] Tsai, Y.-H. H. , Bai, S. , Liang, P. P. , Kolter, J. Z. , Morency, L.-P. , and Salakhutdinov, R. . Multimodal transformer for unaligned multimodal language sequences. Jun 2019. URL `http://arxiv.org/abs/1906.00295v1`.

[133] Uijlings, J. R. , Sande, K. E. V. D. , Gevers, T. , and Smeulders, A. W. . Selective search for object recognition. *IJCV*, 2013.

[134] Uijlings, J. R. R. , Sande, K. E. A. , van de, Gevers, T. , and Smeulders, A. W. M. . Selective search for object recognition. *IJCV*, 2013.

[135] Vaswani, A. , Shazeer, N. , Parmar, N. , Uszkoreit, J. , Jones, L. , Gomez, A. N. , Kaiser, L. , and Polosukhin, I. . Attention is all you need, 2023.

[136] Vinker, Y. , Pajouheshgar, E. , Bo, J. Y. , Bachmann, R. C. , Bermano, A. H. , Cohen-Or, D. , Zamir, A. , and Shamir, A. . Clipasso: Semantically-aware object sketching. *ACM TOG*, 2022.

[137] Vinker, Y. , Alaluf, Y. , Cohen-Or, D. , and Shamir, A. . Clipascene: Scene sketching with different types and levels of abstraction, 2023.

[138] Wang, K. , Wang, Y. , Xu, X. , Liu, X. , Ou, W. , and Lu, H. . Prototype-based selective knowledge distillation for zero-shot sketch based image retrieval. In *ACM MM*, 2022.

[139] Wang, X. , Ang, K. , and Samavati, F. . Sketch-based editing and deformation of cardiac image segmentation. *PRISM*, 2022.

[140] Wei, J. , Bosma, M. , Zhao, V. Y. , Guu, K. , Yu, A. W. , Lester, B. , Du, N. , Dai, A. M. , and Le, Q. V. . Finetuned language models are zero-shot learners. Sep 2021. URL `http://arxiv.org/abs/2109.01652v5`.

[141] Wu, Y. , Kirillov, A. , Massa, F. , Lo, W.-Y. , and Girshick, R. . Detectron2. *https://github.com/facebookresearch/detectron2*, 2019.

[142] Xie, M. , Xia, M. , and Wong, T.-T. . Exploiting aliasing for manga restoration. In *CVPR*, 2021.

[143] Xing, J. , Wei, L.-Y. , Shiratori, T. , and Yatani, K. . Autocomplete hand-drawn animations. *ACM TOG*, 2015.

[144] Xiong, R. , Yang, Y. , He, D. , Zheng, K. , Zheng, S. , Xing, C. , Zhang, H. , Lan, Y. , Wang, L. , and Liu, T.-Y. . On layer normalization in the transformer architecture, 2020.

[145] Xu, P. , Huang, Y. , Yuan, T. , Pang, K. , Song, Y.-Z. , Xiang, T. , Hospedales, T. M. , Ma, Z. , and Guo, J. . Sketchmate: Deep hashing for million-scale human sketch retrieval. In *CVPR*, 2018.

[146] Xu, P. , Joshi, C. K. , and Bresson, X. . Multi-graph transformer for free-hand sketch recognition, 2021.

[147] Xu, P. , Zhu, X. , and Clifton, D. A. . Multimodal learning with transformers: A survey. *arXiv preprint arXiv:2206.06488*, 2022.

[148] Xu, P. , Zhu, X. , and Clifton, D. A. . Multimodal learning with transformers: A survey, 2023.

[149] Xu, R. , Han, Z. , Hui, L. , Qian, J. , and Xie, J. . Domain disentangled generative adversarial network for zero-shot sketch-based 3d shape retrieval. In *AAAI*, 2022.

[150] Yang, J. , Lu, J. , Leel, S. , Batra, D. , and Parikh, D. . Graph r-cnn for scene graph generation. In *ECCV*, 2018.

[151] Yang, S. , Wang, Z. , Liu, J. , and Guo, Z. . Deep plastic surgery: Robust and controllable image editing with human-drawn sketches. In *ECCV*, 2020.

[152] Yelamarthi, S. K. , Reddy, S. K. , Mishra, A. , and Mittal, A. . A zero-shot framework for sketch based image retrieval. In *ECCV*, 2018.

[153] Yu, Q. , Yang, Y. , Song, Y.-Z. , Xiang, T. , and Hospedales, T. . Sketch-a-net that beats humans. Jan 2015. URL `http://arxiv.org/abs/1501.07873v3`.

[154] Yu, Q. , Liu, F. , Song, Y.-Z. , Xiang, T. , Hospedales, T. M. , and Loy, C. C. . Sketch me that shoe. In *CVPR*, 2016.

[155] Yun, S. , Han, D. , Oh, S. J. , Chun, S. , Choe, J. , and Yoo, Y. . Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019.

[156] Zadeh, A. , Chen, M. , Poria, S. , Cambria, E. , and Morency, L.-P. . Tensor fusion network for multimodal sentiment analysis, 2017.

[157] Zareian, A. , Rosa, K. D. , Hu, D. H. , and Chang, S.-F. . Open-vocabulary object detection using captions. In *CVPR*, 2021.

[158] Zeng, Z. , Liu, B. , Fu, J. , Chao, H. , and Zhang, L. . Wsod2: Learning bottom-up and top-down objectness distillation for weakly-supervised object detection. In *CVPR*, 2019.

[159] Zhang, H. , Cisse, M. , Dauphin, Y. N. , and Lopez-Paz, D. . mixup: Beyond empirical risk minimization. In *ICLR*, 2018.

[160] Zhang, L. , Jiang, J. , and Ji, Y. . Smartshadow: Artistic shadow drawing tool for line drawings. In *ICCV*, 2021.

[161] Zhang, X. , Feng, J. , Xiong, H. , and Tian, Q. . Zigzag learning for weakly supervised object detection. In *CVPR*, 2018.

[162] Zhang, X. , Wei, Y. , Feng, J. , Yang, Y. , and Huang, T. . Adversarial complementary learning for weakly supervised object localization. In *CVPR*, 2018.

[163] Zhang, X. , Wan, F. , Liu, C. , Ji, R. , and Ye, Q. . Freeanchor: Learning to match anchors for visual object detection. In *NeurIPS*, 2019.

[164] Zheng, L. , Kang, G. , Li, S. , and Yang, Y. . Random erasing data augmentation. In *AAAI*, 2020.

[165] Zhou, B. , Khosla, A. , Lapedriza, A. , Oliva, A. , and Torralba, A. . Learning deep features for discriminative localization. In *CVPR*, 2016.

[166] Zhou, K. , Yang, J. , Loy, C. C. , and Liu, Z. . Learning to prompt for vision-language models. *IJCV*, 2022.

[167] Zhou, K. , Yang, J. , Loy, C. C. , and Liu, Z. . Conditional prompt learning for vision-language models. In *CVPR*, 2022.

[168] Zhou, X. , Wang, D. , and Krähenbühl, P. . Object as points. *arXiv preprint arXiv:1904.07850*, 2019.

[169] Zhou, X. , Zhuo, J. , and Krahenbuhl, P. . Bottom-up object detection by grouping extreme and center points. In *CVPR*, 2019.

[170] Zhu, X. , Su, W. , Lu, L. , Li, B. , Wang, X. , and Dai, J. . Deformable detr: Deformable transformers for end-to-end object detection. Oct 2020. URL `http://arxiv.org/abs/2010.04159v4`.

[171] Zhu, X. , Su, W. , Lu, L. , Li, B. , Wang, X. , and Dai, J. . Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2021.

[172] Zhu, Y. and Jiang, S. . Deep structured learning for visual relationship detection. In *AAAI*, 2018.

[173] Zitnick, C. L. and Dollár, P. . Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.

.

APPENDIX 1 - WORK PLAN

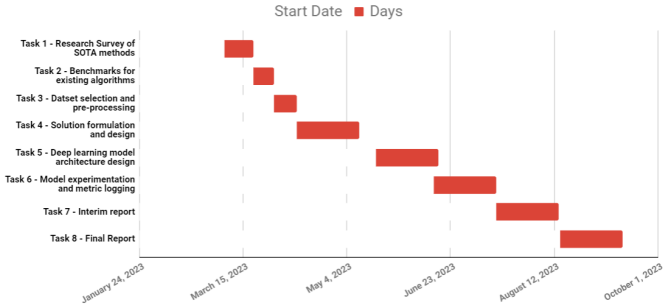

Figure 6.1: Gantt Chart

APPENDIX 2 - TRAINING SUMMARY

All training needs have been fulfilled

6.2 Soft IT Skills

We learnt how to conduct proper literature surveys. Additionally, tools like latex was reviewed and explored in detail

6.3 Professional Skills

Professional Skill	**Training needed Y/N?**	**Current status of training. If not complete, what is the plan to complete it. If training was not needed, state how it had been completed previously.**
Report writing	N	
Research methods, literature reviewing (training provided by the library later in the semester – see timetable)	N	
Oral presentation skills	N	
Time planning/Project planning	N	
Plagiarism training (Go to SurreyLearn and find it in The Student Common Room)	N	

6.4 Specialist Skills

Note you should adapt this table specifically to your project, this template is just given as a guidance.

Specialist Skill	**Training required Y/N?**
Deep learning frameworks	Y

PyTorch and other frameworks were explored

Mathematics – Refreshing or developing knowledge in maths. e.g. Maths Drop In Centre available: http://personal.ee.surrey.ac.uk/Personal/W.Wang/MathsDropInCentre.html The “Casual Drop-In Use” is available to MSc students.	N

Hardware programming, e.g. Arduino, Raspberry PI, FPGA Note that the Electronics and Amateur Radio Society is a great extra curricular way of broadening your skills in firmware programming with devices like Ar-duino. Why not go and join them and attend the courses that students deliver to students? More info at: https://www.ussu.co.uk/ClubsSocieties/societies/ears/Pages/home.aspx	N

Practical skills, e.g. soldering, test and measurement, PCB design You will need to discuss more specif-ically the details with your supervisor if you undertake a practical project of this nature and also you should ensure you have completed a risk assessment for specialist health and safety.	N

Specialist simulation packages. Is there a specialist simulation package that you need to learn use? If you do need to learn a specific simulation package you will certainly need to discuss this with your supervisor and find out how you will learn it.	N

Specialist test and measurement equipment. Some projects involve undertaking a number of practical measurements and may require learning specifically how to use the equipment required for the task.	N

6.4.1 Background state of the art

What are the key words or phrases that you need to search for in the literature regarding the state of the art in relation to your project? You should have discussed this with your supervisor. There may be more than five key words or phrases so you can add more if you need to.

1. sketch
2. FG-SBIR
3. deep learning
4. cross-modal
5. attention

Who are the key scholars in the field where research papers they produce are of relevance to read? You should make note of the key publications you have been reading and have identi-fied as an important resource in completing the literature review of your project that you should have now done.

Key scholars: Prof. Yi-Zhe Song , Ross Girshick , Aditay Tripathi etc.

6.4.2 Theoretical knowledge for your project

What are the key items of theory you need to understand in order to undertake the work rele-vant to your project? How have you progressed with this theory?

1. transformers

2. self-attention
3. vit

The above topics were explaored by reveiwing survey papers, YouTube videos and blogs. Discussions with co-supervisor was held to understand domain knowledge in depth.