Tiangong University

Undergraduate Thesis

Generating Aerial Images using CLIP Latent for Anomaly Detection

使用 CLIP 潜在空间生成用于异常检测的航拍图像

Student ID：	**W2010816010**
Student Name：	**M.W.N.N. MUTHUGALA(那娜)**
Major：	**Artificial Intelligence**
School：	**Artificial Intelligence**
Supervisor：	李云鹏
Professional Title：	讲师
Completion Date：	**June 2024**

摘要

本研究聚焦于 CLIP 和 unCLIP 模型在生成合成航拍图像以改进异常检测方法中的新颖应用。研究评估了这些模型在基于文本描述生成真实航拍图像时的有效性，特别是在实际图像有限或不可用的情况下。随后，生成的图像将使用 YOLOv8 进行训练和优化，以检测异常情况。该过程包括选择适合的图像生成场景、设计文本描述、图像生成、图像预处理、异常标注以及利用数据增强来提升模型的泛化能力。论文探讨了人工生成的图像是否可以用于模型训练以供进一步研究，并研究了在实现图像精度和处理计算需求方面所面临的困难。研究通过详细的测试和分析展示了生成模型（如 unCLIP）和判别模型（如 YOLOv8）对异常检测和航拍图像处理的显著影响。结果强调，unCLIP 生成的航拍图像可以用于模型训练，并显示了进一步研究以提高模型在不同行业中的准确性、效率和相关性的重要性。

关键词：异常检测，航拍图像，生成模型，unCLIP，YOLOv8

Abstract

This study centers on the novel utilization of CLIP and unCLIP models to produce synthetic aerial images for improving anomaly detection approaches. The study assesses the efficacy of these models in producing authentic aerial images based on textual descriptions, particularly in situations when actual images are limited or unavailable. Subsequently, the generated images are trained and optimized using YOLOv8, for the purpose of detecting anomalies. The procedure entails the selection of suitable scenarios for image generation, designing the textual descriptions, image generation, preprocessing of images, the labeling of anomalies, and the utilization of data augmentation to enhance the generalization of the model. The thesis explores if artificially generated images can be used in model trainings for further research studies and the difficulties faced in achieving the accuracy of the produced images and handling the computing requirements. The research showcases the substantial influence of generative models such as unCLIP and discriminative models such as YOLOv8 on anomaly detection and aerial image processing, as evidenced by thorough testing and analysis. The results emphasize that the unCLIP generated aerial images can be used for model trainings and shows significance of further study to enhance the accuracy, efficiency, and relevance of models in different industries.

Keywords: Anomaly Detection, Aerial Imagery, Generative Models, unCLIP, YOLOv8

Table of contents

Chapter 1 Introduction

1.1 Background and Significance

Aerial imaging[1] has greatly transformed various industries[2], including urban planning, agriculture, environmental monitoring, and disaster response. Aerial Images provide a broad view of terrestrial settings, allowing for advanced resource management, planned urban development, and effective catastrophe mitigation techniques. Anomaly detection is a crucial use of aerial imaging, involving the identification of deviations from normal conditions, such as unauthorized buildings, deforestation, or agricultural illnesses. It is important because it enables immediate responses, which in turn promote environmental preservation, improve public safety, and minimize economic losses.

The Contrastive Language Image Pretraining model (CLIP)[3][4] demonstrates exceptional proficiency in understanding and creating visual representations based on written explanations, having been trained on a wide range of images and text obtained through the internet. The capacity to accurately identify and reproduce intricate visual details using text cues makes it highly promising for the field of aerial data analysis.

Applying CLIP to aerial anomaly detection goes beyond improving technological efficiency. It represents a fundamental change in approach towards more flexible, precise, and cost-efficient monitoring systems. These systems are essential for adjusting to the rapid changes in urban landscapes and natural surroundings. By utilizing CLIP's sophisticated image synthesis and identification abilities, researchers can overcome conventional obstacles such as the expensive nature of acquiring aerial images for training anomaly detection monitoring systems and the complicated process of evaluating extensive datasets.

1.2 Anomaly Detection in Aerial Imagery

In aerial imaging, anomaly detection refers to finding unnecessary, unidentified objects, occurrences, or observations in the picture data that substantially differ from the typical patterns or characteristics[5]. This process is crucial in various applications, including emergency management, agriculture, urban planning, and environmental monitoring. Aerial imagery's distinct viewpoint provides a clear hint that is unmatched by ground-level data collecting, making it a priceless tool for quickly and accurately identifying anomalies over wide geographic areas [6].

1.2.1 Historical Context

Anomaly detection in aerial imagery has its roots in manual analysis for land surveying and military reconnaissance [7][8] . Historically, analysts manually reviewed aerial photographs to identify potential threats or resource-rich areas. This meticulous work provided foundational insights into the strategic value of aerial imagery. However, as aerial imaging technology advanced, the volume of collected data grew exponentially, necessitating a shift towards automated approaches.

The transition from manual to automated techniques paralleled broader advancements in computer vision and machine learning. In their review of photogrammetry and remote sensing, Colomina and Molina [9] trace this evolution through the emergence of unmanned aerial systems (UAS). These systems enabled more frequent and expansive data collection, while innovations in image processing helped analysts identify anomalies like unauthorized structures or agricultural pests more efficiently.

Manual annotation processes for crowdsourcing, as highlighted by Mooney and Corcoran [10] , were initially used to mark urban features, providing valuable input for training anomaly detection algorithms. As computer vision techniques evolved, Zhang et al. illustrated how Markov Random Fields could be applied to aerial imagery to identify intricate patterns more accurately.

Lary et al. [11] further emphasize the increasing importance of machine learning in managing the vast amounts of data gathered from aerial observations, which allowed for more nuanced identification of abnormalities over extensive geographic regions. This development has been particularly beneficial for applications like emergency management, agriculture, and environmental monitoring.

Today, the historical shift from manual to automated anomaly detection reflects a broader trend towards leveraging sophisticated machine learning models to analyze and interpret the massive datasets generated by high-resolution aerial imagery. The integration of these technologies has revolutionized anomaly detection, offering rapid and accurate identification across diverse geographic and thematic domains.

1.2.2 Techniques and Approaches

Originally, anomaly detection in aerial images utilized traditional image processing methods such as thresholding, edge detection, and segmentation [12]. These techniques proved effective for basic anomaly identification where the anomalies had distinct features from their backgrounds. As the field progressed, machine learning brought new tools into play, including decision trees, Support Vector Machines (SVM), and k-nearest neighbors (KNN) [13]. These models offered more flexibility in detecting anomalies using examples, though they often struggled with high-dimensional data and required extensive labeled datasets.

The significant advancement in anomaly detection came with the application of deep learning, specifically through Convolutional Neural Networks (CNNs) [14]. CNNs are adept at extracting detailed information from images automatically, making them particularly effective for identifying detailed and subtle anomalies in aerial data. By training on large datasets, CNNs can discern various types of anomalies, from environmental issues to infrastructure damage, significantly enhancing the detection process.

1.2.3 Challenges

Several obstacles remain in the way of anomaly detection in aerial images, regardless of recent developments. The huge volume of data that needs to be processed, which demands a lot of computer power, is one of the main problems. Achieving high accuracy is further complicated by the range of anomalies and their frequently subtle character. The process is made more difficult by the fact that aerial images have a variety of backgrounds from which models must differentiate between real abnormalities and natural variations in the landscape.

1.2.4 Recent Innovations

Recent developments have focused on enhancing anomaly detection's effectiveness and accuracy in response to these difficulties. There may be less need for big annotated datasets thanks to techniques like transfer learning, which adapts a model learned on one job for another. Similarly, to effectively use unlabeled data, unsupervised and semi-supervised learning algorithms have been investigated.

1.2.5 Significance and Applications

It is impossible to overestimate the importance of enhancing anomaly detection in aerial images. It makes it possible to keep an eye on unauthorized building and allocate resources effectively in urban planning. It is essential for identifying deforestation, wildfire damage, and other ecological changes in environmental monitoring. Similar to this, early diagnosis of insect infestations and crop diseases in agriculture can have a big impact on sustainability and food security.

Using state-of-the-art AI technology, anomaly detection in aerial images has progressed from manual inspection to handling complex problems in processing and analyzing large datasets. Notwithstanding the progress made, the sector still faces enormous obstacles concerning accuracy, efficiency, and flexibility in taking on various abnormalities. To overcome these obstacles, research and innovation must continue. They have the potential to make a major impact on several important areas by improving the capacity to observe and react to changes in the environment and human activity from above.

1.3 Generative Models in Computer Vision

Since they provide a method of understanding and reproducing data distribution by creating new data instances that bear resemblance to a given dataset, generative models have emerged as a fundamental component of computer vision. The capacity of these models to generate realistic, high-quality images from learned data distributions makes them especially effective for applications like anomaly detection, augmentation, and image synthesis.

1.3.1 Definition and Types

A group of algorithms known as "generative models" is created to simulate the unsupervised distribution of data. In contrast to discriminative models, which forecast labels based on inputs, generative models can produce new data points. In computer vision, the two main categories of generative models that are commonly employed are:

(1) GANs: Generative Adversarial Networks were first put forward by Ian Goodfellow and his team in 2014 [15]. These GANs consist 2 types of neural networks that are trained simultaneously. These are known as the discriminator and the generator. The discriminator figures out if images are real-images by evaluating them with the real data, and at the same time, the generator generates images in order for them to look like authentic. These 2 neural networks work competitively and continuously. The generator gradually improves its output in response to the discriminator’s feedback.

(2) VAEs: As the name goes, Variational Autoencoders [16] perform based on the principle of autoencoding. Here, the encoder network decompresses into a latent space representation. This is subsequently used by a decoder network to recreate the input. VAEs adds a conditional twist to this procedure by picking from the learned distribution's latent space to produce new data points.

1.3.2 Applications

Many applications in computer vision have been made possible by generative models, including but not limited to image generation [15], for instance, creates new images resembling existing datasets and finds use in game development, digital painting, and data augmentation. Image translation as described in Zhu et al., (2017) enables converting visuals between domains, like shifting a scene from day to night or changing the weather. Super-resolution as described in Ledig et al., (2017) enhances low-quality images into high-resolution versions. Not only that but also, anomaly detection as we follow Hendrycks et al., (2019) identifies unusual patterns by analyzing deviations from the normal distribution, helping detect outliers effectively. These applications demonstrate the broad potential of generative models in computer vision.

1.3.3 Strengths and Limitations

Generative models exhibit strengths and limitations that affect their applicability in various tasks [15]. Strengths include their ability to model complex data distributions, allowing them to generate highly realistic visuals. This capability is particularly valuable in scenarios where augmenting existing datasets is necessary because real-world data is either unavailable or costly to obtain. Furthermore, these models are adept at tasks requiring detailed analysis and manipulation of images, making them versatile tools in fields like medical imaging, entertainment, and autonomous driving.

However, there are notable limitations to consider. The training process for generative models like GANs are computationally intense and require careful tuning of model parameters. This process can be unstable, often leading to the generation of images with artifacts or other unwanted features. Additionally, controlling specific properties of generated images without additional mechanisms or conditioning inputs poses a significant challenge, complicating the generation of images with desired characteristics under constrained conditions.

1.3.4 Recent Advances

Enhancing stability, image quality, and control over generated outputs have been the main goals of recent developments in generative models. The clarity and resolution of generated images have been greatly improved by techniques like the progressive growth of GANs [17] and the introduction of attention mechanisms [18]. Moreover, these models are now more applicable to applications demanding fine control over generated material thanks to conditional generative models, which produce data based on specified conditions or qualities.

In the future, generative model research will focus on creating models with more user-friendly control methods and less processing power to produce complex, high-fidelity visuals. The combination of generative models with other AI fields, such as natural language processing, is an interesting new direction that promises to make it possible to create visuals from written descriptions and vice versa.

1.4 Current Research Status at Home and Abroad

1.4.1 International Research Status

Internationally, the application of the CLIP [3][4] and its derivative models like unCLIP in image synthesis and analysis has attracted considerable attention within the AI research community. While these models have been explored extensively for various tasks such as text-to-image generation [1], art creation [19], and synthetic data generation for training machine learning models, specific applications targeting the generation of aerial images for anomaly detection have not been documented. Although CLIP’s capabilities in understanding and generating images from textual descriptions are well-established, its adaptation for creating aerial imagery specific to anomaly detection remains unexplored, presenting a novel area of research.

1.4.2 Domestic (China) Research Status

Similarly, in China, while there is a robust momentum in utilizing advanced AI technologies like CLIP for enhancing image recognition tasks and improving surveillance systems, the specific application of unCLIP for generating aerial images to detect anomalies is yet to be explored. Chinese research has primarily focused on leveraging AI for urban planning, environmental monitoring, and enhancing agricultural outputs through satellite and drone imagery analysis. However, the potential of unCLIP to synthesize aerial images that could be used to train anomaly detection algorithms represents an untapped opportunity, aligning with the national priority of integrating more AI into critical monitoring and planning operations.

1.5 Main Research Objectives & Structure of the Thesis

1.5.1 Research Objectives

This research project’s primary goal is to increase the capacity of the aerial image analysis with the help of unCLIP model and other trending technologies. The primary objectives are designing and implementing an integrated framework which includes the image generation concepts like CLIP with complementary technologies like object detection and instant segmentation, proving that AI generated images are useful in the image analysis and image processing fields, promote AI research in aerial imagery analysis, bridging gaps between artificial intelligence, computer vision, aerial imagery, urban planning, disaster response and other relevant fields facilitating collaborative experiments and implementing collaborative projects to solve challenges in those fields, and provide resources for the research community which will encourage further innovations and researches in the field of aerial imagery.

1.5.2 Structure of the Thesis

(1) Chapter 1: Introduction

Introduces the research topic, its background, significance, and the current state of research in anomaly detection and generative image synthesis. This chapter also outlines the challenges involved in anomaly detection using aerial images and sets up the thesis structure.

(2) Chapter 2: The unCLIP Method

Provides a comprehensive overview of the CLIP and the specifics of the CLIP and unCLIP models. It also includes the theory of data generation, training methods and finetuning unCLIP.

(3) Chapter 3: Image Generation Principles

Describes the text-based image generation principles, technologies like instant segmentation, prompt engineering and it also discusses about the image to text generation method.

(4) Chapter 4: Dataset Preparation and Training

Discusses how the anomaly detection model is built and trained. It starts with the setting up the environment for the image generation, image generation, describing the selection criteria for anomalies to create quality image data. The chapter also covers fine-tuning the image generation parameters for better data realism. Using the YOLOv8 framework and Roboflow's tools, the dataset is prepared, annotated, and accurately labeled. Lastly, it describes the training configurations and parameters and shows how the custom-trained YOLOv8 model effectively predicts anomalies.

(5) Chapter 5: Experiment Analysis

This chapter explores the final results and outcomes. It starts by analyzing the generated images and discusses the evaluation metrics used to assess the model's performance. Then provides key observations and insights, challenges encountered and the main findings.

(6) Chapter 6: Conclusion and Future Work

Concludes the thesis by reflecting on the research achievements and the impact of this work on the fields of artificial intelligence, computer vision, and aerial imagery analysis.

This structured approach ensures a comprehensive examination of generative models like CLIP and unCLIP in the context of aerial image analysis for anomaly detection, aiming to advance both the academic field and practical applications.

1.6 Summary of this Chapter

This thesis chapter explores how the CLIP model can be integrated with aerial imaging for anomaly detection, emphasizing its potential impact across sectors like urban planning, agriculture, and environmental monitoring. The work highlights how CLIP's image generation and analysis from textual descriptions can improve monitoring systems in terms of precision, flexibility, and cost-efficiency. It traces the evolution of anomaly detection in aerial imaging from manual to automated methods using machine learning, noting the challenges in data analysis and computational demands.

Generative models like GANs and VAEs are examined for their strengths in creating realistic images, despite training limitations. The research also compares global and Chinese efforts, identifying the untapped potential of CLIP in aerial anomaly detection and setting the objectives to build an integrated framework for anomaly detection and interdisciplinary collaboration. The thesis provides a comprehensive review, methodologies using CLIP and unCLIP, experimental validation, and a conclusion, advocating for generative models in aerial imagery analysis to improve anomaly detection and better manage environmental and urban changes.

Chapter 2 The unCLIP Method

2.1 The CLIP Model

The Contrastive Language–Image Pre-training model or as we all call it the CLIP has set a new standard in artificial intelligence which will affect all the other industries in a positive manner [3]. This methodology fosters a natural bridge between visual material and language phrases and greatly advancing machines' ability to interpret and interact with images through written descriptions.

2.1.1 Background and Development

The inception of CLIP stems from the desire to create a model that can understand images in much the same way humans do—through the descriptive power of language. Unlike traditional models that require extensive labeled datasets for each specific task, CLIP is trained on a diverse mix of internet-sourced images and their accompanying text descriptions [20]. This novel training methodology allows CLIP to extrapolate across a diverse array of visual concepts without explicating task-oriented training, emphasizing its adaptability and extensive practicality. Figure 2-1 shows that CLIP pre-trains both an image encoder and a text encoder to predict which images correspond to which texts in the dataset. This pairing ability allows CLIP to function as a zero-shot classifier. By converting all the classes in a dataset into captions like "a photo of a dog," we can identify the caption that CLIP predicts to be the best match for a given image, effectively determining the image's class [3]

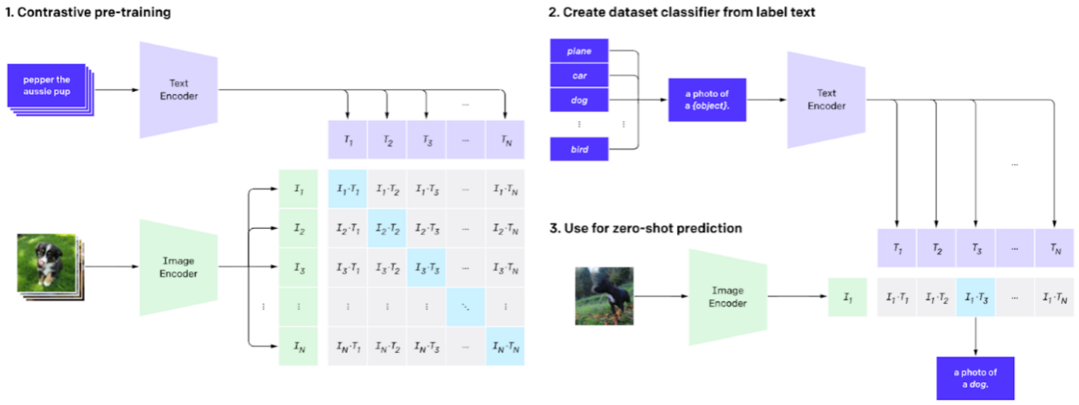

1. Contrastive pre-training
pepper the aussie pup
Text Encoder
Image Encoder
T_1 T_2 T_3 ... T_N
I_1 I_2 I_3 ... I_N
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
$I_2 \cdot T_1$ $I_2 \cdot T_2$ $I_2 \cdot T_3$ ... $I_2 \cdot T_N$
$I_3 \cdot T_1$ $I_3 \cdot T_2$ $I_3 \cdot T_3$ ... $I_3 \cdot T_N$
$I_N \cdot T_1$ $I_N \cdot T_2$ $I_N \cdot T_3$ ... $I_N \cdot T_N$
2. Create dataset classifier from label text
plane
car
dog
bird
a photo of a {object}.
Text Encoder
T_1 T_2 T_3 ... T_N
3. Use for zero-shot prediction
Image Encoder
I_1
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
a photo of a dog.

Figure 2-1 CLIP pre-trains image and text encoders to match images with relevant captions, enabling it to classify images without additional training.

2.1.2 Architecture and Mechanics

The core of CLIP's architecture is centered around two main components: an image encoder and a text encoder. Both encoders independently process images and textual descriptions, translating them into vector representations within the same shared multidimensional space, as described by Radford et al. (2021). This design enables direct comparisons between visual and textual data, which is crucial for associating images with their descriptive text.

Image Encoder: The image encoder f_{img} is often implemented as a convolutional neural network (CNN), optimized for image processing tasks. It received an input image I and returns a high-dimensional vector representation V:

$$V = f_{img}\ (I)$$

(2-1)

The resulting vector V encodes the essential visual features of the image.

Text Encoders: The text encoder f_{txt}, typically a transformer-based model optimized for natural language processing, receives a textual description T as input and outputs a vector representation t.

$$t = f_{txt}\ (T)$$

(2-2)

This vector t represents the semantic meaning of the textual description in a structured high-dimensional space.

Contrastive Learning: CLIP employs a contrastive learning approach to bring together vectors representing matching images and texts while separating those representing non-matching pairs. The similarity between an image vector V and a text vector t is measured using cosine similarity.

$$sim\ (V, t) = \frac{V.t}{\parallel V \parallel \parallel t \parallel}$$

(2-3)

Where $V.t$ is the dot product, and $\parallel V \parallel$ and $\parallel t \parallel$ are the respective vector magnitudes.

The training objective is to maximize similarity between the matching pairs and minimize similarity between non-matching pairs. This objective is achieved through a contrastive loss function.

$$L = -\log \frac{\exp\,(sim(V,t)/T}{\Sigma_{t\prime\epsilon\mathcal{T}} \exp\,(sim(V,t')/T} \tag{2-4}$$

Where, $\mathcal{T}$ is a batch of text descriptions containing both matching (t) and non-matching (t') pairs. And, T is a temperature parameter that scales similarity scores.

According to Radford et al. (2021), this loss function allows the model to refine its cross-modal association capabilities, leading to accurate matching of images and text. The result is a powerful framework that facilitates accurate image-text retrieval and zero-shot learning.

2.1.3 Core Features and Benefits

CLIP is known for its flexibility, SOTA performance, efficiency and robustness. Its training architecture allows it to adapt seamlessly across a broad spectrum of activities without the need for task-specific customization. This adaptability is underscored by its ability to frequently match or even surpass the performance of models specifically trained for tasks like image classification, object detection, and instant segmentation (Radford et al., 2021).

One of the core advantages of CLIP is its efficiency. Unlike traditional models that require large, carefully annotated datasets, CLIP leverages vast amounts of unlabeled data. This approach significantly speeds up the training process, allowing for rapid deployment and scaling. Furthermore, CLIP's robustness is particularly notable. It demonstrates strong resilience and consistent performance across diverse tasks and settings, efficiently handling the classification and identification of images based on textual descriptions. This robustness extends to its ability to perform well even in challenging and varied environments, showcasing its utility in practical applications (Radford et al., 2021).

These features make CLIP a preferred choice for developers and researchers seeking an AI model that combines ease of use with powerful, broad-spectrum capabilities. The model's ability to understand and process information across modalities underscores a significant advancement in machine learning, pushing the boundaries of what AI systems can achieve in visual and textual understanding.

2.1.4 Applications

CLIP's broad capabilities make it exceptionally versatile across various domains by bridging the gap between textual descriptions and visual content. Its proficiency in zero-shot learning enables it to classify images into new categories using only textual descriptions as guidance, demonstrating a profound understanding of visual and textual semantics and allowing it to adapt to new tasks without additional training (Radford et al., 2021). It enhances image search by interpreting natural language queries, providing contextually accurate results through semantic understanding of search terms (Radford et al., 2021). In content creation, CLIP revolutionizes the field by generating photorealistic and artistic images quickly based on text prompts, empowering designers and creators to produce visually compelling content with ease. The model's understanding of text-to-image relationships supports accurate text-to-image generation, where images closely reflect their textual descriptions (Ramesh et al., 2021). CLIP also excels at image retrieval, swiftly identifying images that match textual queries, making it invaluable for managing and accessing large datasets. Moreover, its adaptability extends to image captioning, where it generates descriptive captions that convey image content, particularly aiding accessibility and content management (Radford et al., 2021). These applications showcase CLIP's transformative potential in how machines interpret and interact with visual and textual information, revolutionizing industries from media to automated systems.

2.1.5 Challenges and Ethical Considerations

Although CLIP has shown a lot of promise in a number of areas, there are still issues and moral questions that need to be resolved before it can be used responsibly (Radford et al., 2021).

Bias and Representation: As with many AI models, one of the main issues with CLIP is the possibility that it will pick up biases from the training set. Because CLIP is trained on a wide range of text and image pairs from the internet, it may unintentionally pick up on and reinforce the biases seen in this material. It is imperative to take proactive steps to detect, evaluate, and eliminate biased data from the training set in order to reduce this risk. This also entails making sure that the model's applications continue to be just and equal for all demographic groups and do not perpetuate preconceived notions (Bender et al., 2021).

Complexity in Interpretation: Another significant challenge is the complexity inherent in understanding the decision-making processes of CLIP. The model's ability to associate text with images through a black-box approach can make it difficult for users to interpret why certain associations are made. This opacity necessitates ongoing research efforts focused on making AI operations more transparent and interpretable. Developing methods to elucidate how CLIP processes and responds to inputs can help in building trust and reliability in its applications across sectors (Lipton, 2018).

Addressing these challenges involves a combination of technical improvements and ethical governance to ensure that CLIP and similar AI tools are used in a manner that benefits society while minimizing potential harms.

2.2 Introduction to unCLIP

unCLIP is a noteworthy development in the ongoing evolution of AI and its application to the understanding and production of visual content. It builds on the fundamental principles of CLIP model, expanding CLIP's capabilities beyond image and text correlation to enhanced image analysis and generation. This chapter explores the overview of unCLIP, its architecture, and a comparative analysis with CLIP, emphasizing how unCLIP differs from and advances upon prior models in image analysis and generation.

2.2.1 Overview of unCLIP

unCLIP represents a novel approach that leverages the strengths of CLIP's dual-encoder framework, which processes images and texts separately to understand their correlation. However, unCLIP introduces additional components and mechanisms that enable not only the interpretation of images through textual descriptions but also the generation of images that accurately match given text inputs. This progression marks a significant leap towards creating more intuitive and interactive AI systems capable of bridging the gap between visual and linguistic domains.

The architecture of unCLIP is designed to understand and translate textual descriptions into corresponding images, a process facilitated by its four main components: the Text Encoder, Prior, Decoder and Image encoder. Here, we delve into the roles of these components and the overall architecture process of unCLIP.

(1) Text Encoder

The process starts with the Text Encoder which processes input textual descriptions and transforming them into a high-dimensional vector space. By converting text into embeddings, the Text Encoder enables the model to understand and represent the semantic content of the textual descriptions computationally. This encoder typically utilizes transformer-based architectures, known for their effectiveness in handling sequential data and capturing the contextual nuances of language.

(2) Prior

The embeddings generated by the text encoder are then utilized by the prior. The Prior acts as a bridge between the encoded textual and image embeddings, guiding the image generation process based on the textual description. It leverages the embeddings generated by the Text Encoder to inform the model of what features or elements should be present in the generated image. Essentially, the Prior helps in aligning the textual description with the visual content, ensuring that the generated images accurately reflect the described scenarios or objects.

(3) Decoder

The generative basis of unCLIP is the Decoder. Using the guidance provided by the Prior, the decoder generates images that match the input textual descriptions. It creates visuals that match the input textual descriptions by converting the high-dimensional embeddings into pixel values. Different generating models, each with specific advantages in picture synthesis, can be used by the decoder. These models include diffusion models, variational autoencoders (VAEs), and generative adversarial networks (GANs).

(4) Image Encoder

Finally, the Image Encoder processes the generated images. Its primary function is to encode images into the same high-dimensional vector space as the text embeddings. This encoding allows for the direct comparison and alignment of visual and textual information within the model. This step ensures that the generated images accurately reflect the described scenarios or objects and enables further refinement if necessary. The Image Encoder usually employs convolutional neural networks (CNNs) or Vision Transformers (ViT) to capture complex visual patterns and features effectively.

The Figure 2-2 shows that Text input is processed by the Text Encoder, which converts it into embeddings. These embeddings are then sent both to the Prior for further processing and directly to the Image Encoder for generating images. Similarly, image input is handled by the Image Encoder, which creates embeddings that are forwarded to the Prior and also directly to the Text Encoder for text generation. The Prior plays a crucial role in integrating and guiding these embeddings, which it then directs to the Decoder. Depending on the initiated flow, the Decoder is responsible for producing either images or text. This setup ensures a versatile and bidirectional flow of information, allowing for the generation of both visual and textual outputs from corresponding inputs.

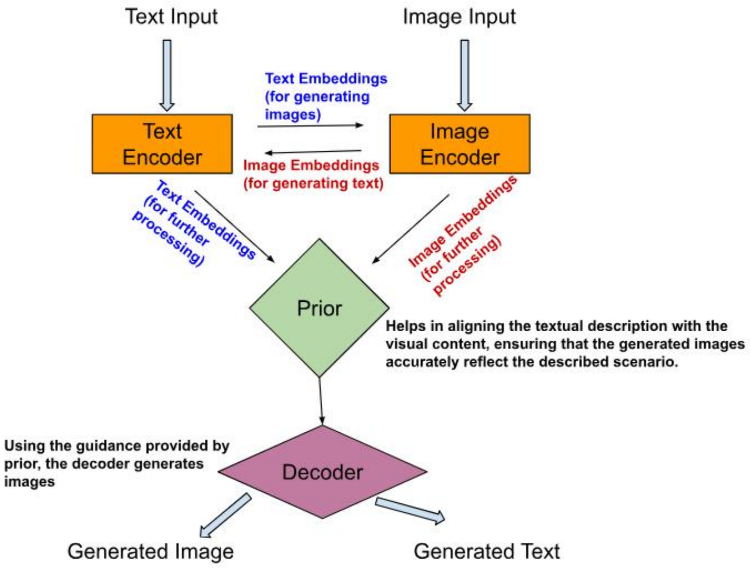

Text Input
Image Input
Text Embeddings (for generating images)
Text Encoder
Image Encoder
Image Embeddings (for generating text)
Text Embeddings (for further processing)
Image Embeddings (for further processing)
Prior
Helps in aligning the textual description with the visual content, ensuring that the generated images accurately reflect the described scenario.
Using the guidance provided by prior, the decoder generates images
Decoder
Generated Image
Generated Text

Figure 2-2 Process Flow Chart of the unCLIP model operation

2.2.2 CLIP Vs unCLIP

Although CLIP was revolutionary in its capacity to comprehend and classify images using written descriptions, its main purpose was to serve as an analytical tool. CLIP has exceptional performance in situations such as zero-shot learning, where it can accurately categorize images into classes that were not included in its training data. Nevertheless, CLIP lacks the inherent ability to create visuals.

unCLIP, on the other hand, extends these capabilities by incorporating image generation, thus transforming the model from a purely analytical tool into a creative one. The generative aspect of unCLIP allows it to produce images that do not merely exist within its training dataset but can be conceptualized through natural language inputs. This advancement not only enhances the model's utility in applications requiring content creation but also deepens our understanding of the intricate relationship between visual and textual information.

2.2.3 Improvements and Differenciations

unCLIP introduces significant enhancements over its predecessors, integrating several key improvements. First, it maintains the robust image analysis capabilities of CLIP, enabling nuanced, context-aware analysis through detailed textual descriptions. The addition of a generative component allows unCLIP to synthesize images from text, unlocking diverse applications across art, design, and educational content creation (Ramesh et al., 2022). Furthermore, unCLIP's architecture is engineered for deep contextual understanding, facilitating the interpretation of complex relationships between text and images to produce coherent, relevant visuals. Its interactive ability to generate images from natural language prompts allows users to directly guide content creation, enriching human-AI interaction. By integrating image generation within CLIP's proven framework, unCLIP advances machine comprehension and creativity, offering valuable insights into human cognition and enhancing AI's role in creative domains (Ramesh et al., 2022).

2.3 Introduction to Data Generation using unCLIP

The process of dataset generation using unCLIP is a crucial step in developing synthetic aerial imagery for anomaly detection. unCLIP's integrated generative capabilities enable the creation of diverse and realistic datasets tailored to specific anomaly scenarios. This is achieved by interpreting detailed textual prompts [4] that describe various normal and anomalous aerial conditions, such as "aerial view of a forest area showing signs of illegal logging" or "overhead image of a flood-affected region with submerged vehicles."

Effective textual prompts are essential for driving unCLIP's generative process [21]. These prompts are crafted to accurately represent the desired anomalies and include detailed descriptions of landscape features, contextual information about the environment, and various perspectives from different angles and altitudes. Once designed, these prompts are input into unCLIP's Text Encoder, which converts them into semantic embeddings. The generative process is guided by these embeddings, with the Prior component ensuring alignment with the semantic content of the prompts, and the Decoder synthesizing images that visually correspond to these descriptions. This may include iterative refinement to closely match the images with the textual descriptions.

After the images are generated, they undergo a crucial annotation and labeling phase where each generated image is accurately labeled based on various anomalies. This annotated dataset then serves as a training set for anomaly detection models, facilitating the training of models that are effective and generalizable to real-world conditions. The dataset undergoes continuous refinement through training, validation, and iterative adjustments to the textual prompts and image generation process, enhancing the quality and relevance of the data for anomaly detection.

The figure 2-3 shows a clear visualization of the steps involves in generating a dataset using unCLIP.

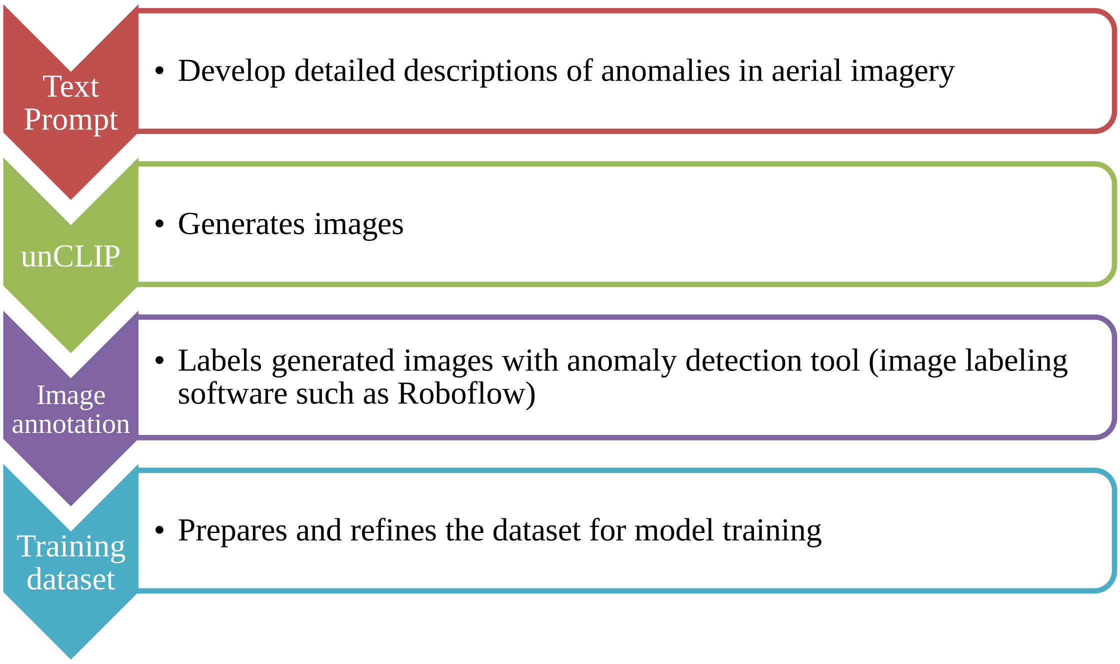

Text Prompt
• Develop detailed descriptions of anomalies in aerial imagery
unCLIP
• Generates images
Image annotation
• Labels generated images with anomaly detection tool (image labeling software such as Roboflow)
Training dataset
• Prepares and refines the dataset for model training

Figure 2-3 Process flowchart for the dataset generation using unCLIP

2.4 Training Method for unCLIP Network

unCLIP extends the capabilities of CLIP by adding a generative component to the framework. This allows not only the encoding of images and text into a shared embedding space but also the generation of one domain from another: either images from text or text from images.

2.4.1 Training Method

The unCLIP network operates on a training dataset consisting of image and text pairs $(x|y)$, where x is the image, and y is the corresponding caption. The training process involves two main components [3].

(1) Prior ($P(z_i|y)$): This component generates CLIP image embeddings z_i conditioned on the text captions y.

(2) Decoder ($P(x, z_i|y)$): This component generates images x conditioned on the CLIP image embeddings z_i, and optionally on the text captions z_i.

2.4.2 Overview of Training Components

Inverted images are produced from their CLIP image embeddings by the decoder, while a generative model $P(x|y)$ of the image embeddings themselves is learned by the prior. $P(x|y)$ for images x given captions y is formulated as:

$$P(x|y) = P(x, z_i|y) = P(x|z_i, y)P(z_i|y)$$

(2-5)

This model is based on the chain rule where z_i is a deterministic function of x. The process involves sampling z_i from the prior conditioned on y, and then sampling x from the decoder using z_i.

2.4.3 Decoder Architecture

The decoder architecture in this system utilizes diffusion models to generate images that are conditioned on CLIP embeddings. Drawing on the principles set out by Nichol et al. (2021), this architecture has been specifically modified to enhance performance. It achieves this by integrating CLIP embeddings into the same space as the timestep embeddings, directly adding them for more effective processing. Additionally, to expand the scope of text conditioning, four extra tokens derived from the CLIP embeddings are concatenated to the outputs of the text encoder. This expansion provides a broader context for image generation. While the text conditioning pathway is intended to grasp subtleties that might be missed by the CLIP model alone, its practical effectiveness has shown to be somewhat limited, indicating areas for potential refinement in future iterations of the model.

2.4.4 Prior Models

The prior in unCLIP is designed to produce a CLIP image embedding from a given caption. Two models are explored for the prior that produces z_i from captions y:

(1) Autoregressive (AR) Prior: This prior transforms the latent space vector z_i into a sequence of discrete codes predicted autoregressively conditioned on the

caption y. The autoregressive prior uses a sequence prediction method, leveraging a Transformer model with a causal attention mask. This setup ensures that the generated codes are strictly influenced by the preceding elements and the associated text.

(2) Diffusion Prior: Models z_i as a continuous vector using a Gaussian diffusion model conditioned on y. The diffusion prior operates through a process that gradually denoises a random vector, guided by the conditioned textual information to form a coherent image embedding.

2.4.5 Loss Function and Optimization

The primary loss function used in the diffusion prior of unCLIP is designed to minimize the mean-squared error between the predicted and actual image embeddings. For the diffusion prior [3], the model predicts the un-noised z_i directly, focusing on mean-squared error loss:

$$L_{prior} = \mathbb{E}_{t\sim[1,T],z_i^{(t)}\sim q_t}[\Vert f_\theta\left(z_i^{(t)},t,y\right) - z_i \Vert]^2$$

(2-6)

This loss function focuses on ensuring that the embeddings produced by the prior are as close as possible to the true embeddings that would result in accurate image generation.

unCLIP is optimized using the Adam optimizer with corrected weight decay and momentum parameters $\beta_1 = 0.9$. This optimization strategy is particularly effective for training models that involve complex interactions between multiple components, such as the generative and discriminative elements in unCLIP.

2.5 Finetuning unCLIP

This chapter outlines the fine-tuning strategies employed to enhance the unCLIP model for generating aerial images specifically tailored to anomaly detection. Aerial imagery, characterized by its varied landscapes and complex patterns, requires precise and contextually accurate image synthesis. The fine-tuning process focuses on adapting unCLIP to effectively handle these challenges, improving its capability to produce detailed and relevant images that aid in detecting anomalies from aerial perspectives.

2.5.1 Domain-Specific Fine-tuning

The model is fine-tuned using a curated dataset comprising aerial images paired with descriptive texts that highlight both normal and anomalous features. This domain-specific dataset enables the model to learn the distinct characteristics of aerial imagery, such as variations in terrain, urban layouts, and natural phenomena. To enhance the model's robustness and its understanding of aerial images, several augmentation techniques are employed. Image augmentation techniques, such as random cropping, rotation, and scaling, simulate different altitudes and angles, reflecting the variability in real-world aerial surveillance. Additionally, text augmentation through paraphrasing and the inclusion of diverse terminologies related to aerial and anomaly descriptors strengthens the model's textual comprehension (Dosovitskiy et al.,2020; Vaswani et al., 2017).

2.5.2 Adjustment of the Loss Function

The contrastive loss function is fine-tuned to improve the alignment between the aerial images and their corresponding text descriptions by increasing the weight of positive samples, thereby reinforcing correct associations and enhancing the model's learning efficacy. Additionally, multi-task learning integrates loss functions from auxiliary tasks such as object localization and semantic segmentation, enabling unCLIP to not only generate images but also to contextualize and identify specific elements within those images, which is crucial for detailed anomaly detection (He et al., 2016; Liu et al., 2021).

2.5.3 Architectural Enhancements

The number and configuration of the Transformer layers in both the text and image encoders are adjusted to handle the complex and detailed descriptions typical in aerial imagery, ensuring deeper processing and richer feature extraction. Additionally, incorporating state-of-the-art convolutional architectures within the image encoder enables better extraction and understanding of intricate image details, which is essential for identifying subtle anomalies in aerial views (Srivastava et al., 2014; Krogh & Hertz, 1992).

2.5.4 Regularization and Optimization Techniques

Implementing enhanced regularization techniques such as dropout and L2 regularization helps prevent overfitting, which is particularly important given the complex nature of aerial image backgrounds and anomaly patterns. Also, employing dynamic learning rate adjustments, such as scheduled decay and cyclical learning rate methods, optimizes training efficacy based on real-time validation feedback, ensuring consistent learning progress (Smith, 2017).

2.5.5 Leveraging Meta-Learning and Transfer Learning

Leveraging pre-trained models from related fields, such as satellite imagery analysis, accelerates the model's adaptation to aerial anomaly detection by utilizing existing knowledge bases to enhance text and image embeddings. Meta-learning approaches are utilized to enable the model to quickly adapt to new, scarce, or evolving aerial image data, which is crucial for applications where data variability is high and anomaly occurrences are rare or subtle (Tan & Le, 2019; Finn, Abbeel, & Levine, 2017; Shin et al., 2016).

2.6 Summary of this Chapter

Chapter 2 provides a comprehensive overview of the CLIP model and introduces its extension, unCLIP, which adds image generation capabilities to CLIP's ability to associate images with text. While CLIP excelled in tasks like zero-shot learning and semantic image search, unCLIP enhances it further by incorporating components like the Text Encoder, Image Encoder, Prior, and Decoder, allowing it to generate images from natural language prompts. The training process involves refining these components using image-text pairs and employing techniques such as contrastive loss tuning and multi-task learning to improve the alignment of images with text. With these advancements, unCLIP bridges the gap between visual and linguistic domains, revolutionizing AI by enabling the generation of visual content based on verbal descriptions. This breakthrough has broad applications, particularly in image generation and anomaly detection in aerial imagery, demonstrating its significant potential across various fields.

Chapter 3 Image Generation Principles

3.1 Text-based Image Generation Principles

Natural language descriptions are translated into visual content using a process called text-based image generation. This technology makes use of developments in artificial intelligence, namely in computer vision and natural language processing (NLP). Here, we talk about the fundamental ideas that support the development of images from textual inputs.

(1) Learning from Large Datasets

Text-to-image models are typically trained on extensive datasets consisting of image and text pairs. These datasets teach the models to understand and correlate the contents of textual descriptions with visual elements [22]. The quality and diversity of these datasets significantly influence the model's ability to generate accurate and contextually appropriate images.

(2) Encoder-Decoder Architecture

Most text-to-image producing models use an encoder-decoder framework [23], which is made up of an image decoder and a text encoder. Text is converted by the text encoder into a high-dimensional space that contains all of the semantic information included in the input text, including linguistic subtleties like grammar and context. This semantic text encoding is subsequently taken by the image decoder, which transforms it into visual data. Upscaling from simple forms and patterns to more intricate and detailed images is what this process entails. To improve the clarity and detail of the generated images, several refinement processes are usually necessary.

(3) Cross-Modal Understanding

A crucial principle in text-based image generation is the ability of the model to understand and translate between different modalities - text and visual content (Mirza, M., & Osindero, S. (2014). Conditional Generative Adversarial Nets). This involves mapping complex descriptions to visual attributes like color, shape, texture, and spatial relationships.

(4) Conditional Generation

The generating process is dependent on the text input. Through controlled generation, where users can direct the content of the images through their descriptions, the model learns to adapt the visual output to fit the details of the written prompt (Mirza, M., & Osindero, S. (2014). Conditional Generative Adversarial Nets.).

(5) Generative Adversarial Networks (GANs)

Many text-to-image models are based on Generative Adversarial Networks (GANs), which are based on the simultaneous training of two neural networks [7][8][15][24]. In this configuration, the discriminator network seeks to distinguish between these artificially generated images and actual real photos, while the generator network creates images intended to closely imitate real images. Through this adversarial process, the generator continuously improves, and the discriminator's feedback is essential for informing and improving the generator's ability to produce images that are more realistic.

(6) Attention Mechanisms

When producing specific parts of the image, attention methods are frequently used in both the encoder and the decoder to concentrate on relevant text parts. This is especially helpful for complicated images when the text describes various aspects of the image in separate sections.

(7) Loss Functions

The training of text-to-image models involves specific loss functions that help in measuring how well the generated images correspond to the text descriptions [25][26]. Commonly used loss functions include contrastive loss (to align text-image pairs correctly) and adversarial loss (to improve the realism of generated images).

(8) Multi-Stage Generation

Models frequently use a multi-stage process where an initial low-resolution image is generated and then gradually refined in order to generate high-resolution images. This method enhances the quality of the output and aids in the management of computational resources.

3.2 Instant Segmentation

In this chapter, we delve into the process of instant segmentation using YOLOv8, a critical step towards achieving precise anomaly detection in aerial images generated by unCLIP. Following the generation of 200 photorealistic images depicting various anomalies, the dataset was meticulously prepared for segmentation. This preparation involved the use of Roboflow, a comprehensive platform designed to enhance the efficiency and accuracy of labeling and segmenting data within images. The choice of instant segmentation over traditional object detection methods is also discussed, emphasizing its significance for this project.

3.2.1 Labeling and Segmenting Anomalies with Roboflow

Roboflow's robust suite of tools significantly enhances the labeling process for anomaly detection in aerial imagery, as it ensures that each anomaly within the generated images is accurately identified and annotated [27]. The process begins with the streamlined uploading of a dataset comprising 200 images into the Roboflow platform, which supports bulk image import and makes the dataset easily accessible for further processing. Once uploaded, the anomaly labeling process commences. Unlike object detection, which uses bounding boxes, instant segmentation involves annotating each detected anomaly with a pixel-level segmentation mask.

In addition to labeling, Roboflow provides several preprocessing options to enhance the dataset quality before segmentation. These steps include resizing images to standardize dimensions across the dataset, normalizing pixel values to increase training efficiency, and applying image enhancement techniques like rotation, flipping, and scaling. These enhancements not only improve the robustness and diversity of the dataset but also bolster the model's ability to generalize to new data, which is crucial for the accurate detection of anomalies.

3.2.2 Choice of Instant Segmentation

The decision to use instant segmentation instead of traditional object detection was based on the project's specific needs. Instant segmentation excels in precision by providing pixel-level classification rather than just bounding boxes, making it particularly valuable for detailed anomaly analysis in aerial images, where accurate boundaries are crucial. It also handles complex scenes effectively, disentangling overlapping anomalies or varied terrain. This method allows for an in-depth analysis of anomalies, enabling better quantification and comparison. Furthermore, it provides a semantically rich understanding of the spatial distribution of anomalies, offering a nuanced comprehension of their arrangement and impact.

3.2.3 Implementation with YOLOv8

After the dataset was labeled and preprocessed in Roboflow, it was exported using YOLOv8 for immediate segmentation. The architecture of YOLOv8 is built to effectively manage the demands of rapid segmentation, utilizing its sophisticated capabilities to deliver precise, real-time segmentation results. The implementation specifics were carefully addressed to guarantee that the segmentation appropriately reflects the labeled anomalies and to improve performance, including adjusting YOLOv8's parameters for this particular purpose.

One important aspect of this project is the immediate segmentation procedure using YOLOv8, which is made possible by the careful preparation and labeling of the dataset in Roboflow. The study pushes the limits of anomaly identification and environmental monitoring by utilizing the most recent developments in AI by selecting immediate segmentation for the thorough investigation of aerial abnormalities.

3.3 Prompt Engineering

Prompt engineering plays a pivotal role in generative AI, particularly with foundational models like unCLIP, which rely on textual prompts to guide image generation. This process involves designing, refining, and optimizing textual inputs to produce accurate and desirable visual outputs. In the context of aerial image generation for anomaly detection, prompt engineering ensures that generated images precisely align with specific research needs.

Crafting Prompts for Image Generation: The first step involves creating prompts that clearly outline the desired characteristics of the generated aerial images. Detailed descriptions, specifying terrain types, environmental conditions, and potential anomalies, are crucial. For instance, a prompt may detail a seaside landscape post-flood or an agricultural field showing signs of pest infestation. The specificity of the prompt directly influences the relevance and fidelity of the generated images, as it guides the model toward creating images that meet research requirements (Brown et al., 2020).

Refining the Prompts Based on Model Responses: After the initial images are generated, the prompts are refined based on the model's outputs. This iterative process assesses how well the generated images match the intended textual descriptions. If they lack accuracy, the prompts are adjusted by incorporating clearer language, specific qualifiers, or additional contextual details. Adjustments help improve the alignment between textual input and the required visual output for anomaly detection.

Experimenting with Different Wording, Parameters, or Instructions: Experimentation is key in prompt engineering. Adjusting the wording, specificity, and instructional parameters of the texts to see how these variations affect the visual output. For instance, changing "flooded urban area with visible water on streets" to "urban landscape submerged under floodwater, with cars and buildings partially underwater" results in distinct visual emphases. Experimentation helps identify effective prompt structures and styles for eliciting precise responses from the model.

Prompt engineering enhances the quality and utility of generated images, improving anomaly detection accuracy. By systematically crafting, refining, and experimenting with prompts, we could refine the interaction between textual inputs and visual outputs, contributing to better AI methodologies (Ramesh et al., 2021).

3.4 Image to Text

The Image-to-Text conversion within the unCLIP framework represents a transformative approach to interpreting and understanding visual data by converting it into semantically rich textual descriptions. This process is crucial for aerial anomaly detection, providing a detailed means of analyzing and documenting significant features or irregularities detected in aerial images (Ramesh et al., 2021).

The reverse process of "image to text" leverages the dual-encoder architecture of unCLIP to translate generated images back into their most likely textual descriptions. This process maps both images and text into a shared vector space, enabling the retrieval of text descriptions that could have prompted the creation of each specific image (Radford et al., 2021). First, the images are processed through unCLIP's image encoder, transforming them into high-dimensional vector representations. These image vectors are then matched against a pre-existing database of vectors derived from various textual prompts. By identifying the closest matching text vectors, the system retrieves the most probable textual descriptions corresponding to the images. This innovative approach allows for meaningful extraction of relevant textual information, aiding in comprehensive analysis and documentation of visual data.

Figure 3-1 will help visually represent the sequential steps involved in converting the generated images back into their most likely textual descriptions using the unCLIP model, emphasizing the role of each component in this innovative process.

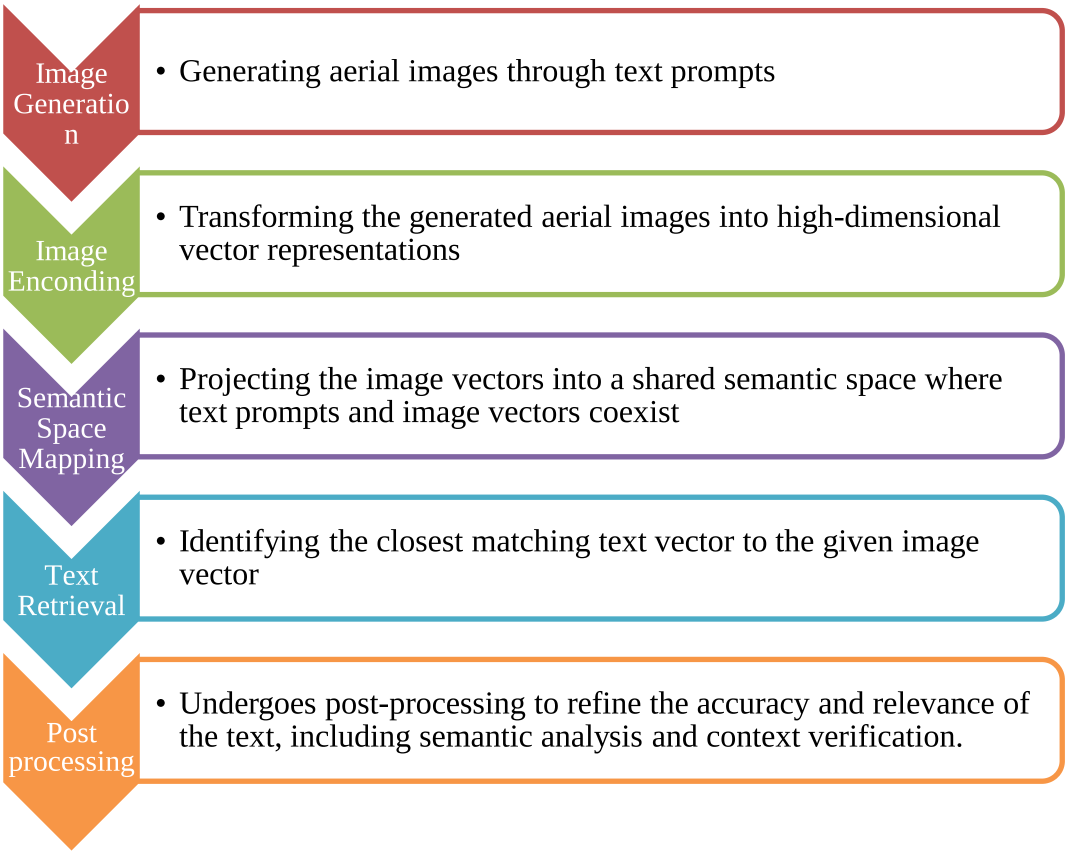


Image Generatio n
• Generating aerial images through text prompts
Image Enconding
• Transforming the generated aerial images into high-dimensional vector representations
Semantic Space Mapping
• Projecting the image vectors into a shared semantic space where text prompts and image vectors coexist
Text Retrieval
• Identifying the closest matching text vector to the given image vector
Post processing
• Undergoes post-processing to refine the accuracy and relevance of the text, including semantic analysis and context verification.

Figure 3-1 Process flowchart for the Image to Text conversion

3.4.1 Encoding Visual Features

The first step involves generating a set of aerial images using the unCLIP model based on carefully engineered prompts that describe various scenarios likely to contain anomalies. These scenarios include descriptions of environmental changes, unauthorized constructions, or disaster impacts. By employing unCLIP, we can create highly detailed and diverse images that reflect these scenarios, allowing us to further explore the model's capability in generating accurate and varied visual data.

$$f_{img} = CLIP_i(\mathrm{x}) \tag{3-1}$$

Where f_{img} is the image feature vector and x are the input image [28].

3.4.2 Semantic Space Mapping

The encoded vectors f_{img} are then projected into a shared semantic space where textual prompts and image vectors coexist. This cross-modal understanding is facilitated by the projection layers within the unCLIP framework, allowing for effective comparison and retrieval tasks.

$$f_{cimg} = W_i \cdot f_{img}$$

(3-2)

Where, W_i is the visual projection layer transforming image features into the CLIP visual embedding f_{cimg} [28]

3.4.3 Retrieval of Text Descriptions

In this step, the semantic vector f_{cimg} is compared against a database of vectors derived from textual descriptions. This comparison identified the closest matching text vector, effectively enabling the model to generate a textual description of the visual content.

$$Description = \arg min_{s \in S} \parallel \cos(f_{ctxt}, f_{cimg}) \parallel$$

(3-3)

Where S represents the set of all textual description vectors f_{ctxt}. [28].

3.4.4 Post-processing and Refinement

To ensure the accuracy and relevance of the generated text, the descriptions undergo post-processing. This phase may involve semantic analysis and context verification to refine the alignment between the text and the visual content. Techniques such as re-ranking based on contextual relevance and combining descriptions are commonly employed to enhance the descriptive quality.

3.4.5 Application in Anomaly Detection

This technology plays a vital role in automated labeling of anomalies detected in aerial images, which is crucial for tasks such as environmental monitoring and urban planning. By providing precise textual descriptions of anomalies, the system aids in the comprehensive analysis and documentation of each incident, thus improving response strategies and planning.

In conclusion, the Image-to-Text conversion capability of unCLIP significantly enhances the interpretation of aerial imagery (Kingma and Welling, 2013). This methodology not only supports current analytical needs but also contributes to a continuous learning process, refining the detection and description capabilities of AI models in aerial anomaly detection. This iterative improvement is pivotal for developing more accurate and context-sensitive aerial imaging analysis systems.

3.5 Summary of this chapter

This chapter provides a comprehensive overview of key concepts and methodologies encompassing text-based image generation, instant segmentation, prompt engineering, and image-to-text conversion. It covers principles such as learning from large datasets, cross-modal understanding, conditional generation, and multi-stage generation. The chapter also explains the rationale and implementation of instant segmentation with Roboflow and YOLOv8 for labeling anomalies, while emphasizing the importance of prompt engineering in crafting, refining, and experimenting with textual prompts. Finally, it details the image-to-text conversion process through unCLIP's dual-encoder architecture, describing how images are processed to retrieve likely textual descriptions via backward propagation, ultimately showcasing applications and implications in generating and analyzing aerial imagery.

Chapter 4 Dataset Preparation & Training

4.1 Data Preparation

As mentioned before, the pivotal first step in this research study of anomaly detection in aerial imagery using unCLIP involves the meticulous preparation of a dataset. This dataset is uniquely crafted to encompass a wide range of anomalies detectable from aerial perspectives, spanning environmental, urban, and infrastructural domains. The generation of this dataset was achieved through the innovative application of unCLIP, a state-of-the-art generative model capable of transforming textual descriptions into photorealistic images. This section delineates the comprehensive process undertaken to create 200 aerial images, detailing the selection criteria for anomalies, the setup and execution of the unCLIP model, and the generation parameters.

4.1.1 Experiment Environment Setup and Model Initialization

To establish the experimental environment and initiate the model for this study, the unCLIP model was accessed and used through Google Colab, a cloud-based Jupyter notebook service that provides an accessible platform for running complex machine learning models without extensive computational resources.

Google Colab was chosen as the platform to leverage its GPU capabilities, which are essential for the computationally intensive process of image generation. The unCLIP model used, identified as kakaobrain/karlo-v1-alpha, is noted for its proficiency in generating high-fidelity images based on textual prompts. The technical steps involved in generating these images were carefully executed to ensure the production of accurate and useful data for anomaly detection.

The technical steps involved in generating images were as follows:

1) Environment Setup:

The necessary Python libraries were installed, including diffusers, transformers, accelerate, and safe tensors. These libraries provide the essential functions and classes required to interact with the unCLIP model and process the generated images.

Table 1: Code Block

```
!pip install diffusers transformers accelerate safetensors
```

The ! at the beginning of the command is a feature specific to Jupyter notebooks and some other interactive environments like in Google Colab Notebooks. It allows you to run shell commands directly from the notebook environment.

2) Model Initialization:

The unCLIP model, specifically "kakaobrain/karlo-v1-alpha", was instantiated using the `UnCLIPPipeline` from the `diffusers` library. To accommodate the intensive computational demands of image generation, the model was configured to utilize CUDA-enabled GPUs, ensuring efficient execution.

Table 2: Code Block

```
from diffusers import UnCLIPPipeline
import torch

pipe = UnCLIPPipeline.from_pretrained("kakaobrain/karlo-v1-
alpha", torch_dtype=torch.float16)
pipe = pipe.to('cuda')
```

During the generation phase, specific parameters were employed to optimize the process. The `torch_dtype=torch.float16` parameter was selected to strike a balance between computational efficiency and the fidelity of the generated images. Additionally, the model was configured to operate in GPU mode `(to('cuda'))` to take advantage of accelerated computing capabilities, which are crucial for processing large-scale image data efficiently.

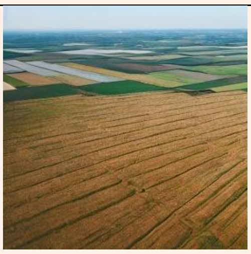
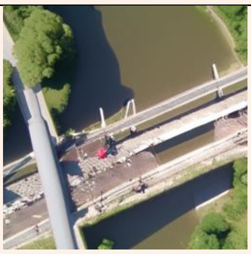
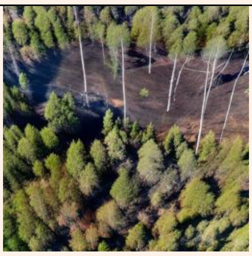
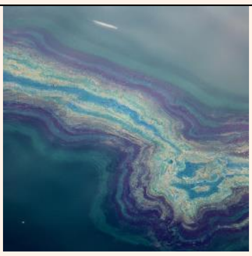
4.1.2 Image Generation

In the image generation phase of the study, over 70 descriptive prompts were carefully crafted to represent a wide array of anomalies observable from aerial perspectives. These prompts covered a diverse range of scenarios, including environmental degradation, infrastructure collapse, and various signs of distress or irregularity within both natural and human-made landscapes. Utilizing the unCLIP model for each prompt, a corresponding image was generated that visually interpreted the described scenario. This process was systematically repeated for all prompts, culminating in the creation of a dataset comprising 200 photorealistic images. This dataset is crucial for training and testing the accuracy and effectiveness of the anomaly detection models.

Table 3: Code Block

```
prompt = “aerial view of a huge land of agriculture with
anomaly”
image = pipe(prompt).images[0]
image
```

Table 4: Generated Images

Prompt	Generated Image
Aerial View of a huge land of agriculture with anomaly	
Aerial photo of a bridge with structural damage after an earthquake	
Aerial view of a forest with a large, unnatural clearing, indicative of illegal logging	
Overhead shot of a river with unusual color patches, suggesting chemical spill	

4.1.3 Selection Criteria for Anomalies

In the selection criteria for anomalies section of the study, the choice of anomalies was based on their relevance to real-world issues that can be observed through aerial imagery. This included environmental threats, both urban and rural infrastructural damages, and signs of ecological imbalances. A key focus was placed on diversity and representativeness to ensure that the dataset encompassed a broad spectrum of anomalies. This approach was intended to enhance the robustness of the subsequent anomaly detection model.

4.2 Fine-tuning Parameters for Better Image Generation

To optimize the quality of the generated images, several parameters were fine-tuned, including:

(1) Image Resolution: Adjustments were made to generate higher-resolution images, ensuring that the details of anomalies are visible and distinguishable.

Table 5: Code Block

`pip.config.resolution = (1024, 1024)`	Set resolution to 1024×1024

`prompt = “Aerial view of a city showing traffic congestion in high detail”`
`high_res_image = pipe(prompt).images[0]`
	Generate High resolution image

(2) Prompt Specificity: The level of detail in the text prompts was fine-tuned to find a balance between generality and specificity, ensuring the model generates images that are diverse yet accurately represent the described scenarios.

Table 6: Code Block

```
general_prompt = “Aerial view of a landscape”
specific_prompt = “aerial view of a winding river
cutting through a dense green forest”

image_general = pipe(general_prompt).images[0]          Generate images
image_specific = pipe(specific_prompt).images[0]
```

(3) Model Parameters: Parameters within the unCLIP model, such as the number of iterations for the diffusion process and temperature settings for creativity and randomness in image generation, were adjusted to achieve the best results.

Table 7: Code Block

```
pipe.config.num_diffusion_steps = 1000           Increase the number of
                                                 diffusion steps.
pipe.config.temperature = 0.8                    Adjust temperature to
                                                 influence randomness.

prompt = “Detailed aerial view of a coastal      Generate images
area showing erosion patterns”
custom_image = pipe(prompt).images[0]
```

(4) Integration with Detection Algorithms: For a comprehensive analysis, the generated images were also processed using object detection algorithms, including a preliminary version of YOLOv8, to identify and classify anomalies within the images. This integration required adjustments to the image processing pipeline to accommodate the format and requirements of the detection algorithms.

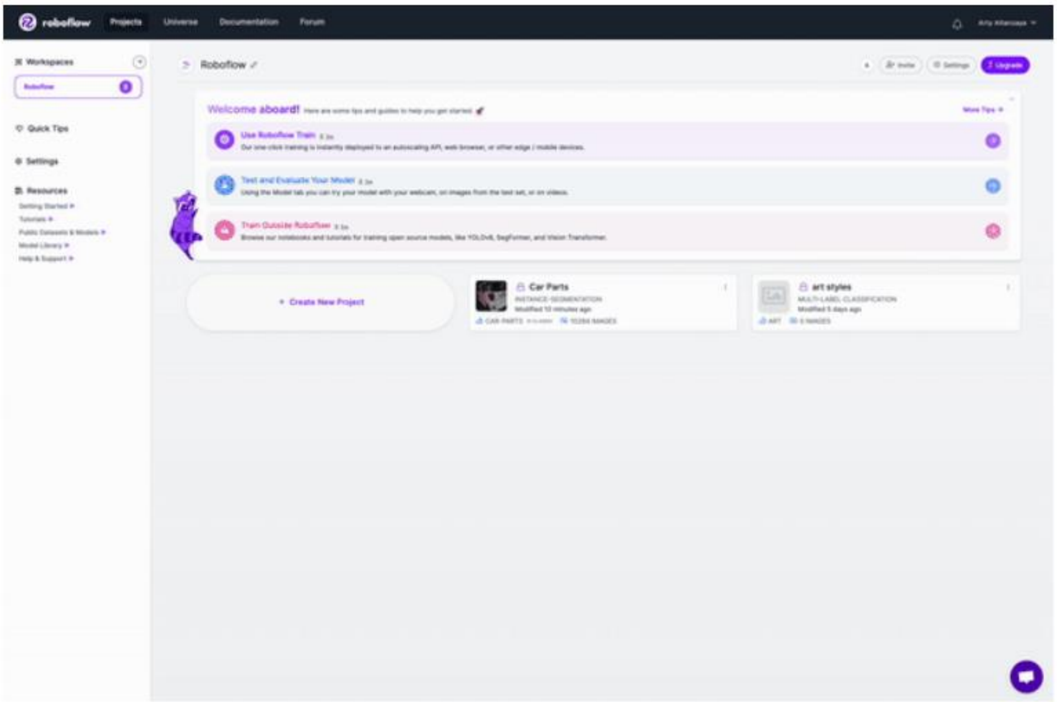
4.3 Training with YOLOv8

In the pursuit of advancing anomaly detection within aerial imagery, this chapter focuses on the training process of YOLOv8 for instance segmentation, executed on Google Colaboratory. This phase is pivotal for refining the model to recognize and delineate anomalies with high precision. The process encapsulates dataset preparation through Roboflow, configuring the model for this specific task, and detailing the training parameters used to achieve optimal performance.

4.3.1 Dataset Preparation with Roboflow

The initial step in this training process involved preparing the custom dataset of 200 generated images depicting various anomalies. Using Roboflow, the annotation and preprocessing of these images were streamlined to suit the requirements of YOLOv8 instance segmentation.

(1) Project Setup: A new project was initiated on Roboflow's platform, categorizing it under "Instance Segmentation". This classification was crucial for ensuring the annotations included precise pixel-level details necessary for this segmentation task.



Here are some tips and guides to help you get started

Our one click training is instantly deployed to an autoscaling API, web browser, or other edge / mobile devices.

Using the Model tab you can try your model with your webcam, on images from the test set, or on videos.

Browse our notebooks and tutorials for training open source models, like YOLOv8, SegFormer, and Vision Transformer.

Figure 4-1 Roboflow Interface

(2) Uploading and Annotating Images: 200 images were uploaded to Roboflow via the web interface. The platform's intuitive annotation tools were employed to label each anomaly within the images.

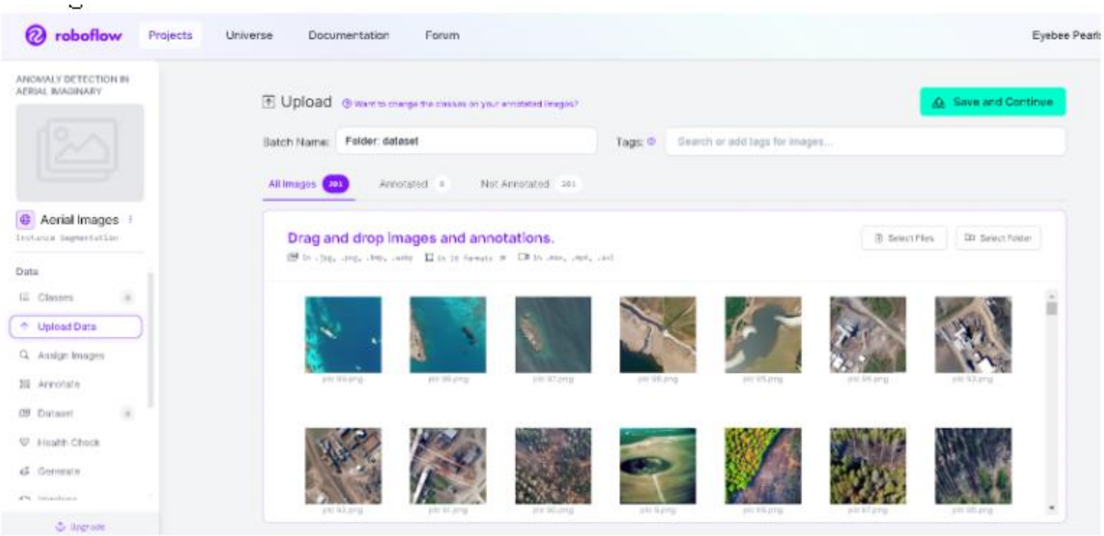


Figure 4-2 Roboflow Interface

(3) Preprocessing and Augmentation: Several preprocessing steps and augmentations were applied to enhance the robustness and generalizability of this model. These included resizing the images to a uniform dimension and applying random rotations and flips to introduce variability.

(4) Exporting the Dataset: Once annotated and preprocessed, the dataset was exported in the YOLOv8 PyTorch format, making it readily compatible with our training environment on Google Colaboratory.

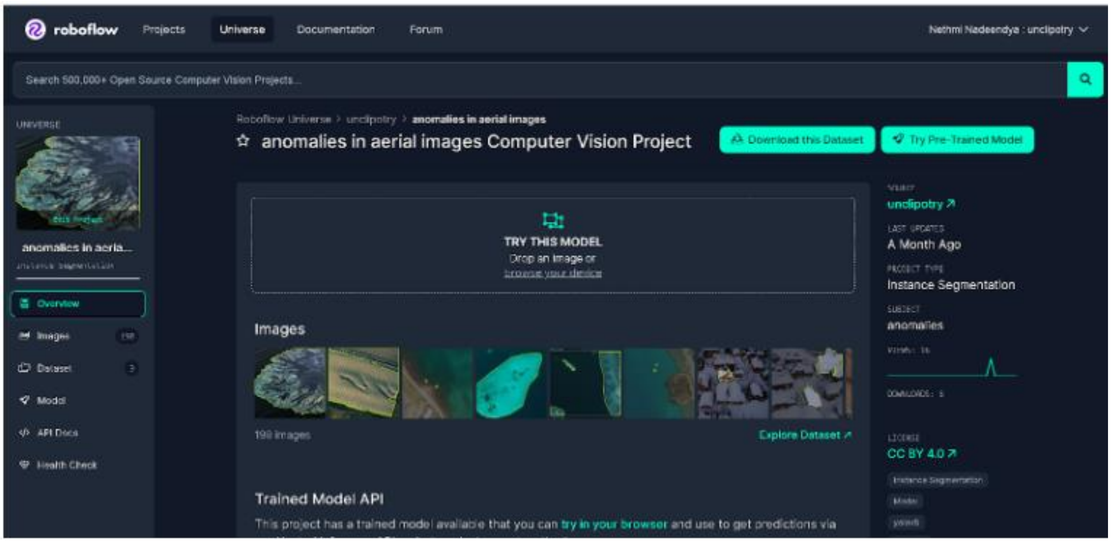


Figure 4-3 Roboflow Interface

4.3.2 Model Configuration & Training Parameters

With the dataset prepared, the transition was made to setting up the YOLOv8 instance segmentation model for training. This process involved several critical steps executed within a Google-Colab notebook, ensuring access to the necessary computational resources.

(1) Installation: The first step involved installing the specific version of Ultralytics required for YOLOv8.

Table 8: Code Block

`!pip install ultralytics==8.0.196`
 	
`from IPython import display`
`display.clear_output()`
 	
`import ultralytics`
`ultralytics.checks()`
	The code installs the ultralytics package, clears the output display, and then runs a system check using the ultralytics package to verify its setup.

(2) Model Initialization: Utilizing the command-line interface (CLI) provided by Ultralytics, YOLOv8 was initialized for instance segmentation. This involved specifying the model configuration, including the path to the custom dataset, training epochs, and image size.

(3) Training: The training command executed within the Colab environment initiated the model's learning process. Our model, `yolov8s-seg.pt`, was trained over a specified number of epochs, with an image size of 640 for optimal balance between performance and speed.

Table 9: Code Block

`%cd {HOME}`
 	
`!yolo task=segment mode=train model=yolov8m-seg.pt data={dataset.location}/data.yaml epochs=25 imgsz=640`
	The code initiates training of the YOLOv8 model in segmentation mode using the specified dataset and configuration file, running for 25 epochs with an image size of 640.

To analyze the outcomes of the training process, such as the confusion matrix, predictions, and validation batch, you can execute the provided code.

Table 10: Code Block

```
Image(filename=f'{HOME}/runs/segment/train/MaskF1_curve.png',
width=600)
```

Displays an image named MaskF1_curve.png from the runs/segment/train directory with a width of 600 pixels.

4.3.3 Predicts with a Custom YOLOv8 Instance Segmentation Model

After training the customized model, you may assess its performance on a different dataset by using the following command in the CLI:

Table 11: Code Block

```
!yolo task=segment mode=val
model={HOME}/runs/segment/train/weights/best.pt
data={dataset.location}/data.yaml
```

Runs the YOLOv8 model in segmentation validation mode using the best model weights from the training run and the specified dataset configuration file.

To execute instance segmentation on new images using your trained model, utilize the following CLI command:

Table 12: Code Block

```
!yolo task=segment mode=predict
model={HOME}/runs/segment/train/weights/best.pt conf=0.25
source={dataset.location}/test/images save=true
```

The code runs the YOLOv8 model in segmentation prediction mode using the best model weights from the training run, with a confidence threshold of 0.25, on images located in the test set directory, and saves the prediction results.

(This code section utilizes a prediction algorithm to identify objects in the test photos and then saves the output images along with bounding boxes and masks.)

4.3.4 Training Observations and Adjustments

Throughout the training process of the YOLOv8 model for instance segmentation, several performance metrics were diligently monitored, and necessary adjustments were made to enhance accuracy and efficiency. A key focus was placed on loss reduction; by observing the decrease in loss over epochs, informed adjustments to learning rates and augmentation strategies were implemented to optimize training outcomes. Additionally, a subset of the dataset was used for validation purposes to evaluate the model’s ability to generalize to unseen data. This step was crucial to ensure that the training effectively prepared the model without it being overly fitted to the training set.

Hyperparameter tuning also played a significant role, with iterative adjustments to learning rates and batch sizes being made to discover the most effective settings for the specific task. This phase of hyperparameter tuning was critical in refining the model's performance and ensuring it was optimally configured for the challenges at hand.

The training of the YOLOv8 model on a custom dataset using Google Colaboratory for computational tasks and Roboflow for dataset management marked a significant milestone. This process not only showcased the practical application of advanced AI models to real-world challenges but also underscored the importance of meticulous dataset preparation and model configuration. As the project progresses, these foundational efforts will be crucial in applying the trained model to detect and analyze anomalies within aerial imagery, pushing the boundaries of what is achievable with AI in environmental and urban monitoring.

4.4 Summary of this Chapter

Chapter 4 provides an in-depth exploration of dataset preparation and training methodologies for anomaly detection in aerial imagery using AI technologies. It covers data preparation by creating a diverse dataset of 200 aerial images depicting various anomalies through unCLIP, highlighting environment setup and image generation via Google Colab with carefully crafted prompts to ensure diversity. The chapter also delves into fine-tuning parameters, like image resolution and prompt specificity, to optimize unCLIP’s performance, while custom training data is integrated with YOLOv8 detection algorithms for improved anomaly distinction. Furthermore, it outlines the training process with YOLOv8 for instance segmentation, detailing dataset preparation, model configuration, and training commands, along with observations, adjustments, and performance metrics that underscore the iterative nature of model refinement. Overall, Chapter 4 emphasizes the meticulous processes required for effective dataset preparation and fine-tuning, demonstrating how cutting-edge AI models are practically applied to tackle real-world environmental and urban monitoring challenges.

Chapter 5 Experiment Analysis

5.1 Generated Images

5.1.1 Notable Characteristics and Patterns

The generated images display notable characteristics and patterns that highlight their effectiveness in anomaly detection. They encompass a broad spectrum of diversity and specificity, with each image tailored to the exact characteristics described in the prompts. For instance, "illegal logging" images clearly depict forest clearings, while "chemical spills" capture distinctive coloration indicative of pollution. The images also showcase exceptional detail and context, offering intricate precision in representing abnormalities alongside their broader environment. For instance, unCLIP adeptly captures both the wide aerial view and minute details in "crop circles indicating pest infestation" or "solar panel farms with damaged panels." The photorealistic quality of the images enhances their utility in simulation training, environmental monitoring, and urban planning, with textures, lighting, and shadows that mimic real aerial photographs. This realism is largely dependent on the specificity and clarity of the input prompts, as more detailed prompts produce more accurate and contextually relevant images. And the same prompt could generate difference images and never produced the same image twice. Additionally, unCLIP-generated images generally exhibit high resolution and clarity, enabling the identification and analysis of aerial anomalies through fine details.

5.1.2 Success Rate of the Generated Images

Defining success criteria for the project involves establishing specific benchmarks to evaluate the effectiveness of the generated images. Several indicators have been used to measure the success rate of the generated images through a structured approach that assesses both quantitatively and qualitatively. Realism is a crucial criterion; the images should closely mimic real-world aerial imagery in terms of quality and detail. Relevance is also essential, as each image must accurately represent the scenario described by its corresponding textual prompt. Lastly, the clarity of anomalies within the images is vital, ensuring that both human observers and AI models designed for anomaly detection can easily discern them. These criteria collectively aid in assessing the performance and utility of the generated images in realistic applications.

(1) Quantitative Evaluation

Here we will carry out an image Quality Assessment by utilizing image quality metrics like Structural Similarity Index (SSIM) and Peak Signal-to-Noise Ratio (PSNR) to assess the realism and accuracy of generated images in comparison to actual aerial photographs.

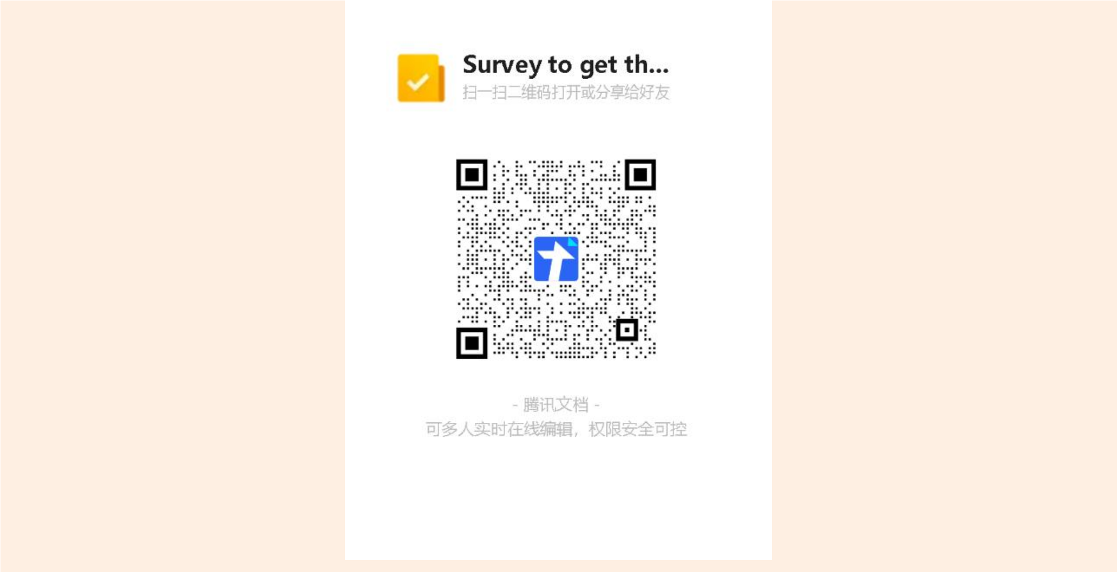
(2) Qualitative Evaluation

Involve humans in evaluating the generated images for their realism, relevance to the prompts, and the clarity with which anomalies are depicted. This could involve a blinded review process as well. In order to measure the overall success-rate involving humans a public survey has been done.

5.1.3 Public Survey

40 humans from various age groups, various fields of specialization, and various nationalities have been involved with this survey to have an understanding of the overall image quality of these generated images.

The survey is done by sending a QR code which directs everyone to a form which has 5 simple multiple-choice questions.



Survey to get th...

扫一扫二维码打开或分享给好友

- 腾讯文档 -

可多人实时在线编辑，权限安全可控

Figure 5-1 The Survey QR code

Table 13: The Survey Questions

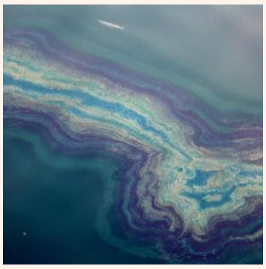
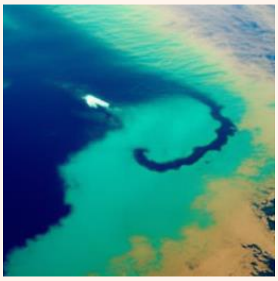
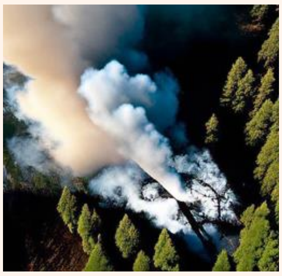
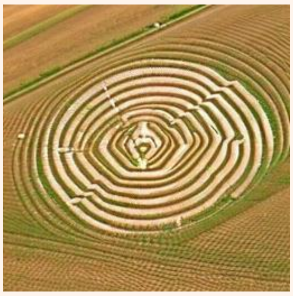
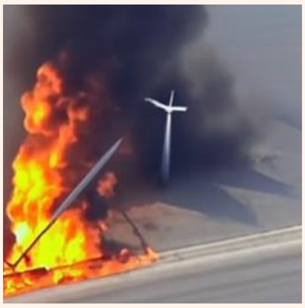
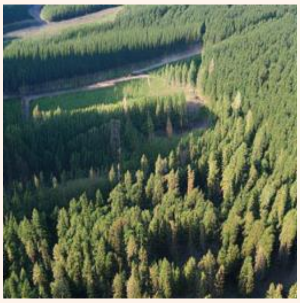
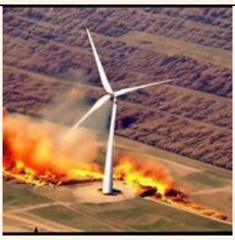
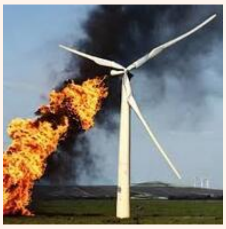
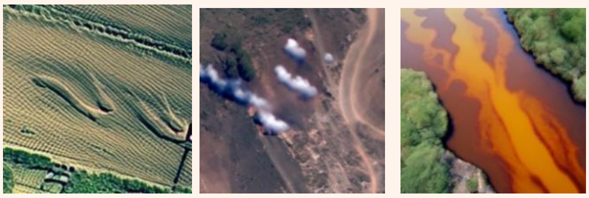
Survey to get the public opinion on the generated images using unCLIP model. 调查获取公众对使用 unCLIP 生成图像的意见	
These aerial images with anomalies (something that deviates from what is standard, normal, or expected.) are generated using OpenAI's unCLIP. Please take a look at these images which are generated by entering text. 这些带有异常的航拍图像（即与标准、正常或预期不符的事物）是使用 OpenAI 的 unCLIP 生成的。请看一下这些通过输入文本生成的图像。	
Image 1- Overhead shot of a river with unusual color patches, suggesting chemical spill 一条河的俯视图，显示出异常的颜色斑块，暗示有化学泄漏。	Image 1 
Image 2 - Drone view of a coastal area with a section of coral reef bleaching. 一片海岸区域的无人机视图，显示出一部分珊瑚礁褪色。	Image 2 
Image 3 - Drone image of a forest with a large area scorched by wildfire. 一片森林下的无人机航拍图像，显示出大面积被野火烧毁。	Image 3 
Image 4 - Aerial image of farmland with crop circles indicating pest infestation. 一片农田的航拍图像，显示出作物圈被害虫侵扰。	Image 4 

Image 5 - Aerial image of a wind farm with one turbine on fire. 一个风力发电场的航拍图像，显示出一个风力涡轮机着火了。	Image 5 
Image 6 - Aerial view of a mining area with unauthorized expansion into protected lands. 一个采矿区域的航拍视图，显示出违规扩张进入受保护的土地。	Image 6 
Question 1: Do you think these generated images look like real images? 你认为这些产生的图像看起来像真的吗？	Yes （是） No（否） Maybe（可能）
Question 2: These generated images closely mimic real-world aerial imagery in quality and detail. 这些生成的图像在质量和细节上与真实世界的航拍图像非常相似。	Yes （是） No（否） Maybe（可能）
Question 3: Each image accurately represents the scenario described by its corresponding textual prompt. 每张图像准确地呈现了其对应的文本提示所描述的情景。	Yes （是） No（否） Maybe（可能）
Question 4: Here, there are 2 images. One is a real image downloaded from the internet. One is an image generated using unCLIP. Pick the generated image. 这里有两张图片。一张是从互联网下载的真实图片。另一张是使用 unCLIP 生成的图像。请选择生成的图像。	 

Question 5: Can you spot the anomalies (Spotting things that don't look right or normal. Eg: illegal buildings, checking on forest cutting, spotting crop diseases early, and seeing how bad natural disasters were) in the following 3 images? 你能发现以下 3 张图像中的异常吗？（发现看起来不正常或不正常的事物。例如：非法建筑物，检查森林砍伐情况，及早发现作物疾病，以及了解自然灾害的严重程度） 	Yes（是） No（不是） Maybe（可能

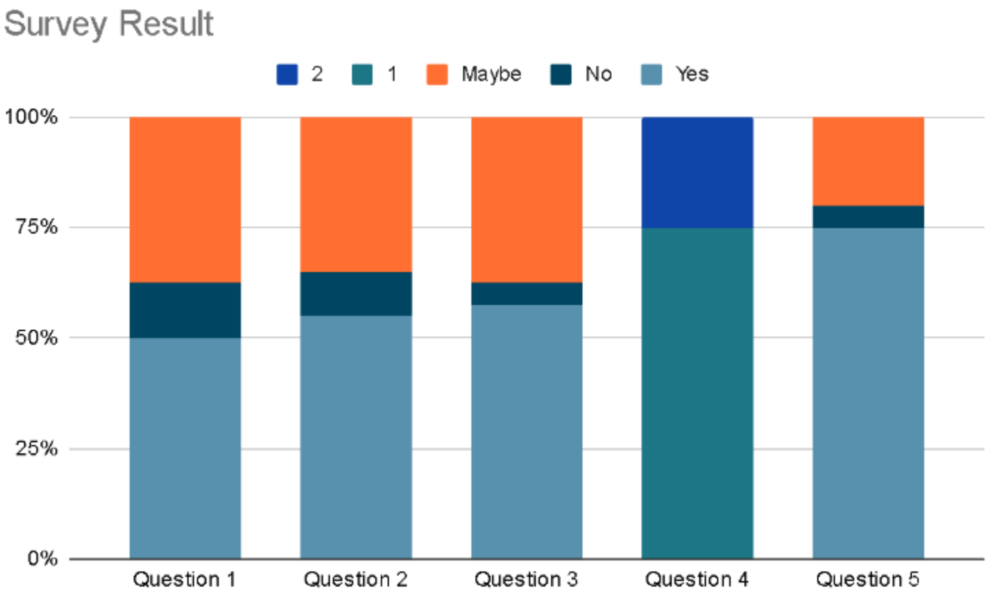


Figure 5-3 The Survey Results

Survey Explanation and Conclusion

The survey conducted to gauge public opinion on aerial images generated using unCLIP model sought to assess the realism, relevance, and clarity of anomalies depicted in these images.

Table 14 - Survey Results

Question/Category	Overview	Survey Results
Realism of Generated Images	Whether the public perceives generated images as realistic.	20 Yes, 15 Maybe, 5 No. unCLIP effectively creates believable images.
Quality and Detail	If participants find the images closely mimic real-world aerial imagery.	22 Yes, 14 Maybe, 4 No. Humans think the generated images are similar to real images.
Accuracy of Representation	Whether images align with their prompts.	23 Yes, 15 Maybe, 2 No. The prompts create relevant imagery.
Real vs. Generated	If participants distinguish real vs. unCLIP generated images.	30 generated, 10 real. Realism is improving.
Anomaly Detection	Can participants spot anomalies in the images?	30 Yes, 8 Maybe, 2 No. Anomalies are identifiable.

The survey results indicate a generally positive reception to the images generated by unCLIP, with the majority of participants affirming their realism, quality, and relevance to the provided prompts. The ability of most participants to spot anomalies within these images underscores the potential utility of unCLIP-generated imagery in applications such as environmental monitoring and urban planning.

However, the presence of "Maybe" responses and the challenge faced by some participants in distinguishing generated images from real ones highlight areas for improvement. Enhancing the photorealism and reducing identifiable discrepancies that set generated images apart from real ones could further improve the utility and acceptance of such AI-generated imagery.

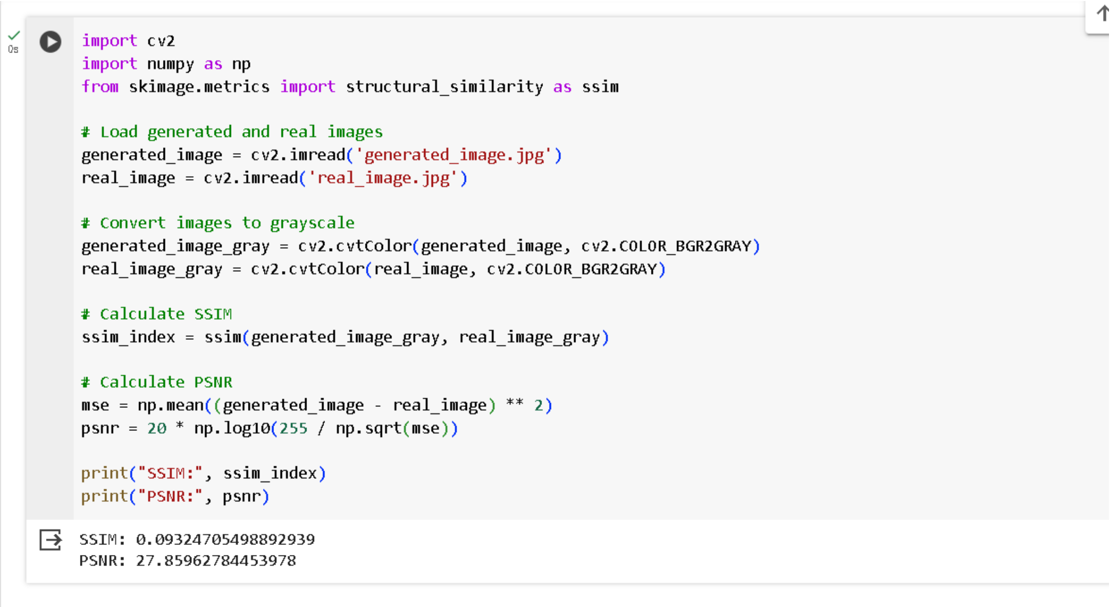
5.1.4 SSIM & PSNR

Metrics like the Structural Similarity Index (SSIM) and Peak Signal-to-Noise Ratio (PSNR) are used to measure how similar two images are to one another and to evaluate the quality of images. This provides a mathematical framework for comparing the accuracy and realism of generated images to those that are actual.

A perception-based model called SSIM takes texture, brightness, and contrast variations between two images into account. The range of SSIM values is -1 to 1, with 1 denoting perfect similarity. A high SSIM score indicates that, in terms of structural integrity and visual composition, the generated image closely resembles the original image.

The ratio of a signal's maximal potential strength to the amount of corrupting noise that degrades the representational fidelity is called PSNR. Usually, decibels (dB) are used to express it. Greater similarity and reduced inaccuracy between the generated and reference images are indicated by higher PSNR values. A high PSNR in the context of image production indicates that the produced image maintains a high level of fidelity in comparison to the source.

Analysis of SSIM and PSNR Results



```
import cv2
import numpy as np
from skimage.metrics import structural_similarity as ssim

# Load generated and real images
generated_image = cv2.imread('generated_image.jpg')
real_image = cv2.imread('real_image.jpg')

# Convert images to grayscale
generated_image_gray = cv2.cvtColor(generated_image, cv2.COLOR_BGR2GRAY)
real_image_gray = cv2.cvtColor(real_image, cv2.COLOR_BGR2GRAY)

# Calculate SSIM
ssim_index = ssim(generated_image_gray, real_image_gray)

# Calculate PSNR
mse = np.mean((generated_image - real_image) ** 2)
psnr = 20 * np.log10(255 / np.sqrt(mse))

print("SSIM:", ssim_index)
print("PSNR:", psnr)
```

```
SSIM: 0.09324705498892939
PSNR: 27.85962784453978
```

Figure 5-4 Code Block

SSIM Score of 0.0932: This low score suggests that the produced images and their real-world equivalents differ significantly in terms of structural substance and visual texture. It implies that although the generated photos might be able to convey the general idea or subject matter of the original photographs, they are very different in little characteristics like texture, brightness, and contrast.

PSNR of 27.859 dB: Depending on the application, PSNR values in image processing might vary greatly, but generally speaking, a PSNR of 30 dB or more is regarded as suitable for image compression. Although it is near this threshold, a score of 27.859 dB denotes a substantial amount of noise or inaccuracy in the generated images when compared to the original ones. It implies that there are some obvious disparities between the created and genuine photos that may be seen with more analysis.

5.2 Evaluation Metrics

The evaluation metrics from the Google Colaboratory training sessions of YOLOv8 for instance segmentation provide insightful data on the performance across 25 epochs.

5.2.1 Key Metrics Overview

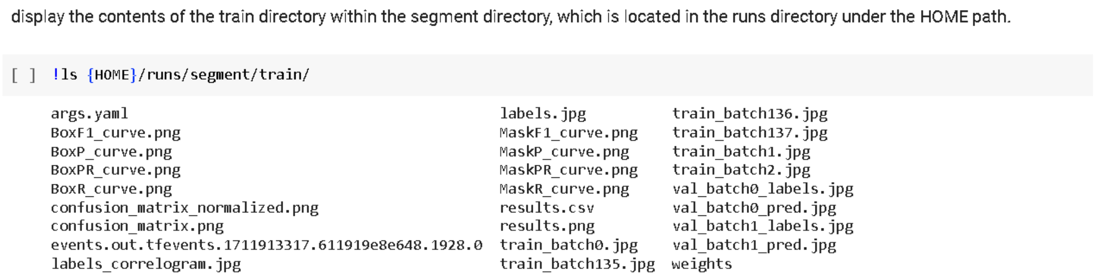
The overview of key metrics provides a way to evaluate how well the YOLOv8 model is performing during its training. These metrics include Box Loss, Segmentation Loss, Class Loss, and Directionally Focused Loss (DFL Loss), which all measure different types of errors in how the model predicts and segments images, classifies objects within them, and focuses on specific directions in the images. These measurements tend to change as the model learns and improves, showing the model's progression and adjustments made to enhance its accuracy in image segmentation and object classification.

Additionally, two important metrics called Precision and Recall are used to further assess the model's performance. Precision measures how accurate the model's predictions are out of all predictions made, while Recall checks how well the model can identify all relevant instances within the data. The metrics mAP50 and mAP50-95 give an overview of the model’s overall performance by calculating the average precision at a 50% threshold and at thresholds ranging from 50% to 95%. Together, these metrics provide a comprehensive view of how effectively and accurately the model processes and analyzes complex images.

5.2.2 Performance Over Epochs:

The initial epochs show a learning phase where the model struggles with higher losses and lower precision and recall, as evidenced by low Box(P) and Mask(P) scores and initial mAP50 and mAP50-95 scores at epoch 1. As training progresses, there is a gradual improvement in most metrics, though fluctuations in loss values and Precision/Recall scores indicate challenges in achieving consistent high accuracy, particularly in distinguishing fine details for instance segmentation. By epoch 25, there is notable improvement in precision, recall, and mAP scores, reflecting the model's enhanced ability to accurately segment and classify instances within the images, with final mAP50 and mAP50-95 scores showing a modest yet significant enhancement in segmentation capabilities.

5.2.3 Visual Illustrations



display the contents of the train directory within the segment directory, which is located in the runs directory under the HOME path.

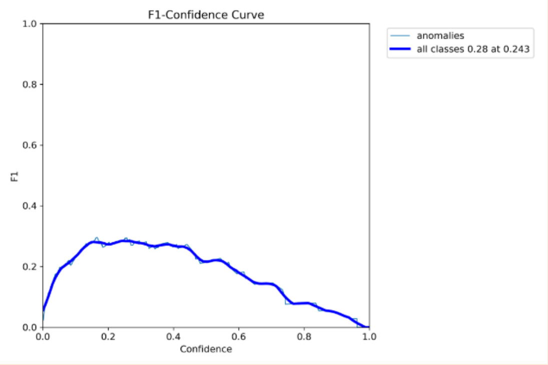
```
[ ]  !ls {HOME}/runs/segment/train/

     args.yaml                                           labels.jpg          train_batch136.jpg
     BoxF1_curve.png                                     MaskF1_curve.png    train_batch137.jpg
     BoxP_curve.png                                      MaskP_curve.png     train_batch1.jpg
     BoxPR_curve.png                                     MaskPR_curve.png    train_batch2.jpg
     BoxR_curve.png                                      MaskR_curve.png     val_batch0_labels.jpg
     confusion_matrix_normalized.png                     results.csv         val_batch0_pred.jpg
     confusion_matrix.png                                results.png         val_batch1_labels.jpg
     events.out.tfevents.1711913317.611919e8e648.1928.0  train_batch0.jpg    val_batch1_pred.jpg
     labels_correlogram.jpg                              train_batch135.jpg  weights
```

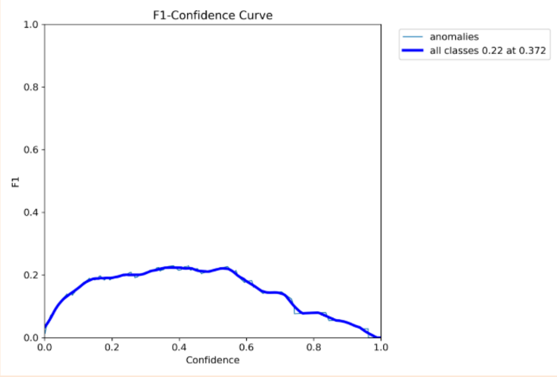
Figure 5-5 Code Block

To visually illustrate the evaluation of the YOLOv8 instance segmentation model effectively, the following images would be most relevant to display:

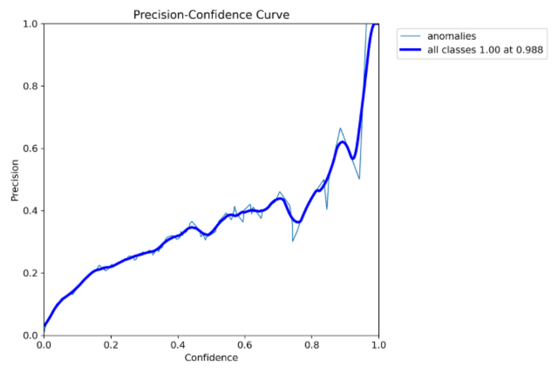
(1) BoxF1 curve



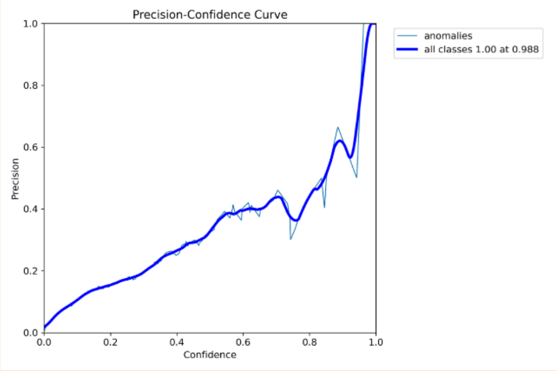
(2) MaskF1 curve



(3) BoxP curve



(4) MaskP curve



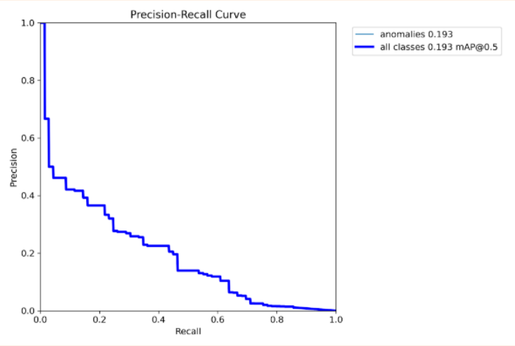
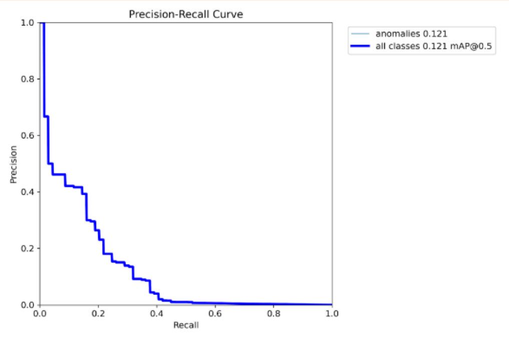
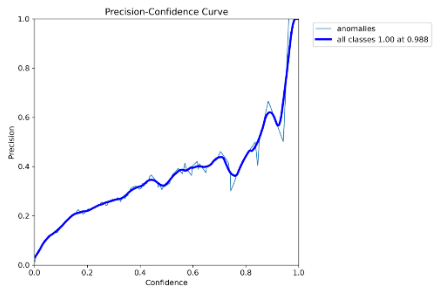
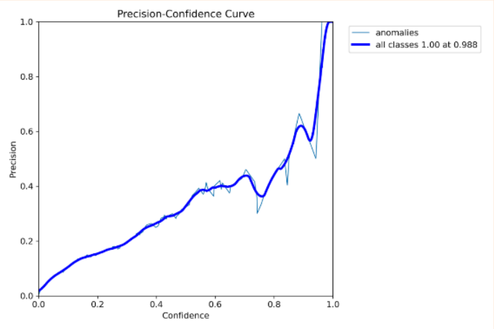
(5) BoxPR curve  	(6) Mask PR curve 
(7) Box R curve  	(8) Mask R curve 

Table 15 - Evaluation Curves

The evaluation metrics depicted in table 15: various graphical formats provide a comprehensive insight into the YOLOv8 model's performance in anomaly detection through instance segmentation. The "BoxF1_curve.png" and "MaskF1_curve.png" both demonstrate the F1 scores—a combination of precision and recall—across varying confidence thresholds. In these plots, the BoxF1 curve reveals the model's cautious approach at higher thresholds with an optimal F1 score peaking before declining, suggesting potential areas for enhancement in model precision and recall. Similarly, the MaskF1 curve shows a trade-off between precision and recall in segmenting anomalies, with the peak indicating the best balance but still displaying a relatively low score, which points to challenges in segmentation accuracy.

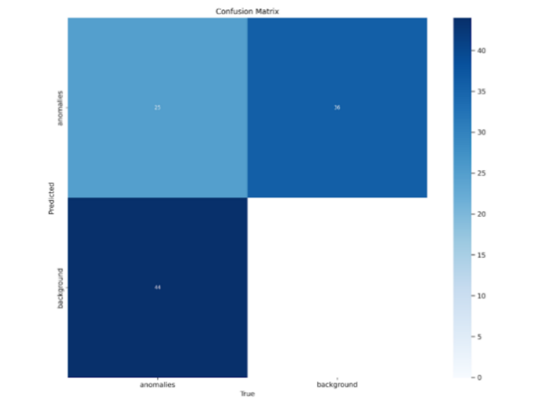
The "BoxP_curve.png" and "MaskP_curve.png" provide insights into precision across different confidence thresholds for bounding boxes and masks, respectively. These curves highlight how precision adjusts with the confidence threshold, generally starting higher at lower thresholds and decreasing as the threshold increases, which is indicative of the model's conservative prediction strategy.

Further, the "BoxPR_curve.png" and "MaskPR_curve.png" display precision-recall curves that illustrate the relationship between these two metrics at various confidence levels. These curves are crucial for understanding the balance between detecting relevant instances and avoiding false positives, which is essential for fine-tuning the model's performance.

Additionally, the "BoxR_curve.png" and "MaskR_curve.png" graphs show recall for bounding boxes and masks over different confidence thresholds, emphasizing the model's capability to capture relevant anomalies at varying levels of stringency in prediction confidence.

Lastly, the "Confusion_matrix.png" and its normalized version "Confusion_matrix_normalized.png" detail the model's classification accuracy across different classes, offering a percentage-based view that helps compare the prediction performance in a balanced way across varied dataset sizes. These visual tools collectively provide a detailed analysis of the model's capabilities and shortcomings, highlighting areas where improvements can be made to enhance accuracy and reliability in detecting and classifying aerial anomalies.

(9) Confusion matrix curve



(10) Confusion matrix normalized curve

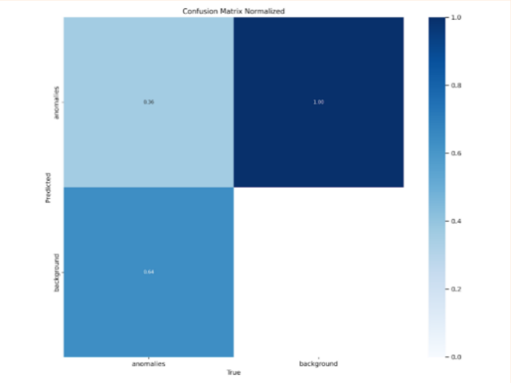


Table 16- Model Evaluation curves

The "Confusion_matrix.png" and "Confusion_matrix_normalized.png" serve as essential tools for evaluating the performance of the classification model used in the study. The confusion matrix visually represents the classification results by displaying the counts of true positives, false positives, true negatives, and false negatives, providing a straightforward depiction of the model's accuracy and areas where it may confuse classes. The normalized confusion matrix complements this by adjusting these values to show proportions, making it easier to compare performance across datasets of different sizes. This normalization is particularly valuable in dealing with unbalanced datasets, as it highlights the model’s performance relative to the prevalence of each class, offering insights into how effectively the model predicts across various categories without the distortion caused by class size discrepancies. These matrices together furnish a comprehensive view of the model’s classification capabilities, pinpointing strengths and weaknesses in its ability to discern and correctly classify different types of anomalies within aerial images.

Figure 5-6 shows the results.png which provides a concise overview of the model's overall performance by visually summarizing important metrics including precision, recall, mAP50, and mAP50-95.

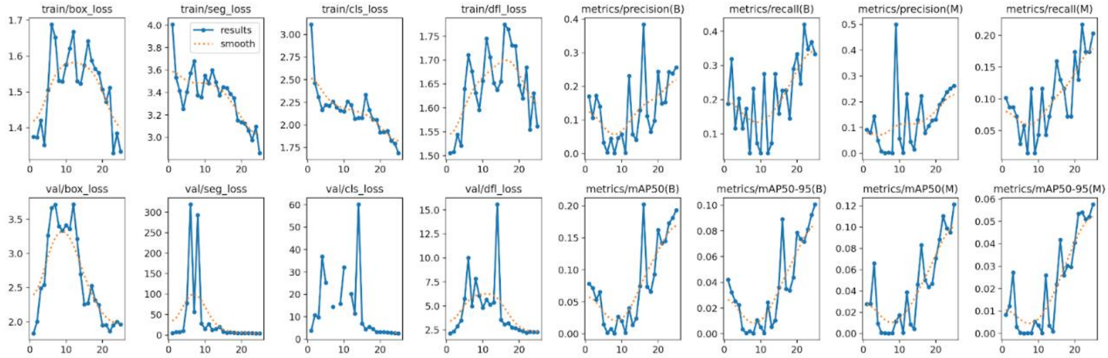


Figure 5-6 Model Performance Results

The training and validation processes for the YOLOv8 instance segmentation model are meticulously documented through a series of metrics that reflect various aspects of model performance. These include box loss, which assesses how closely the model’s predicted bounding boxes align with the ground truth during object detection tasks, observed during both training and validation phases. Similarly, classification and segmentation losses during training and validation reveal the model's accuracy in segmenting and classifying the images correctly.

Additionally, deformation loss, specific to the model's design, is tracked to understand more nuanced aspects of its predictive abilities. Precision metrics for boxes and masks indicate the proportion of correct identifications among all detections, providing insight into the accuracy of the model’s instance segmentation. Meanwhile, recall metrics measure the model's ability to capture all relevant instances, which is crucial for comprehensive anomaly detection.

Mean Average Precision (mAP) metrics for both boxes and masks at the 50% Intersection over Union (IoU) threshold, and more detailed mAP measurements across a range of IoU thresholds (from 50% to 95%), offer a granular view of the model’s performance in detecting and delineating objects across various difficulty levels.

The training process is visually represented through various curves that plot these metrics over epochs, with blue lines indicating raw values and orange dashed lines suggesting smoothed trends for clearer visibility of performance over time. This collection of metrics collectively provides a thorough evaluation of the model's ability to fit the training data and generalize effectively to new, unseen datasets, essential for robust anomaly detection in aerial imagery.

The table 17 given below offers a side-by-side comparison of ground truth labels versus model predictions for a batch of validation images, highlighting the model's segmentation capabilities in real scenarios.

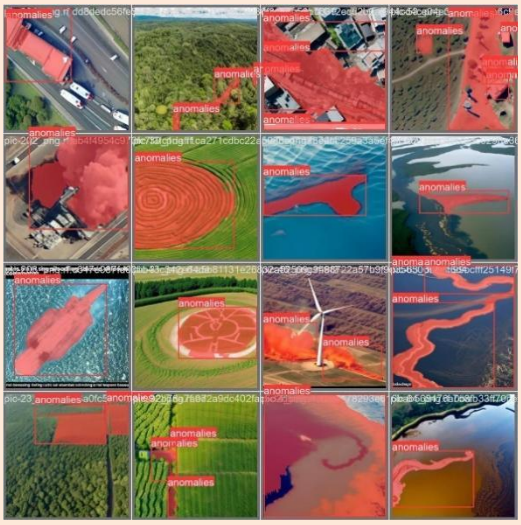
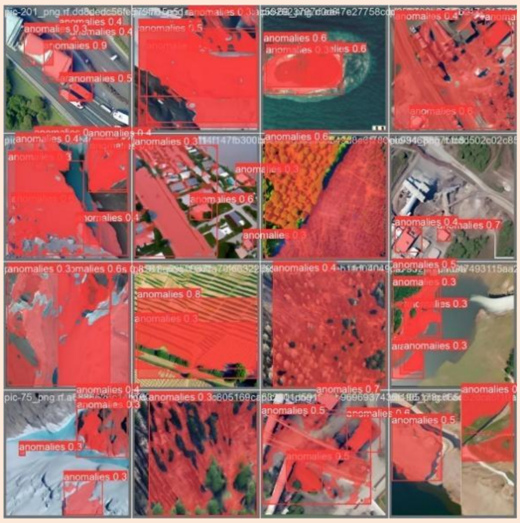
(11) Validation batch 0 labels image	(12) Validation batch 0 predictions image
	

Table 17- Model Evaluation curves

The validation batch 0 labels image and the validation batch 0 predictions image play a crucial role in assessing the performance of the YOLOv8 instance segmentation model on unseen data. The labels image includes precise, manually applied annotations that mark the locations of anomalies within the photographs, often represented through masks or bounding boxes. These annotations may also categorize each anomaly, labeling them with terms like "chemical spill" or "forest fire."

Conversely, the predictions image, named val_batch0_pred.jpg, displays the model's predictions for the same set of images. It shows where the model has identified potential anomalies, marked by bounding boxes or segmentation masks, and each prediction may be accompanied by a confidence score, indicating the model's certainty about each identified anomaly.

The primary purpose of presenting these two types of images side-by-side is to facilitate a direct comparison between the model's predicted outputs and the ground truth labels. This comparison allows for a qualitative evaluation of the model's accuracy and reliability in detecting and classifying anomalies in new data, providing insights into its generalization capabilities and areas for potential improvement.

5.3 Observations and Insights

The evaluation metrics and images from training the YOLOv8 instance segmentation model on Google Colaboratory provide a comprehensive understanding of the model's learning progression and accuracy across 25 epochs. The fluctuations in box, segmentation, class, and DFL loss metrics indicate a typical learning curve, as the model gradually adjusts and improves with more data. Precision and recall metrics also show an upward trend, highlighting improved identification and classification capabilities. Early variance in these values reflects the model's refining understanding of class-defining features. As training progresses, the model's segmentation and classification abilities improve, with mAP50 and mAP50-95 scores indicating enhanced accuracy and generalization. Despite modest improvements in the final epochs, further enhancement remains possible. Initially, higher losses and lower precision/recall are observed, which is expected as the model learns. Optimizing the confidence threshold is crucial, as indicated by F1 score curves for boxes and masks, to balance precision and recall effectively. The val_batch0_labels.jpg and val_batch0_pred.jpg images offer valuable qualitative analysis by visually comparing the model's predictions to ground truths.

5.4 Challenges Encountered

In this sub-chapter, the various challenges encountered during different phases of this research project, such as image generation, segmentation, and model training, are discussed. It also details the strategies implemented to overcome these obstacles to refine the research process and enhance the outputs.

(1) Challenges in Image Generation

Utilizing generative models like unCLIP presented several challenges, particularly in producing realistic and varied images that aligned accurately with the textual descriptions. One major issue was the fidelity of the generated images; initially, some images lacked necessary detail or included artifacts. This was addressed by fine-tuning the generative model's parameters and iteratively refining the text prompts, which led to substantial improvements. Additionally, filters were implemented to exclude images below a quality threshold, ensuring that only high-fidelity images were utilized for training. Another challenge was achieving diversity in the generated anomalies. The dataset often reflected biases inherent in the generative model, which was counteracted by augmenting the dataset with additional prompts that targeted underrepresented anomalies and incorporating varied descriptors and contexts within the prompts to ensure a comprehensive dataset.

(2) Challenges in Segmentation

Segmentation posed significant challenges, particularly in accurately delineating the contours of anomalies within images. Early model iterations often struggled with precise boundary segmentation, tending to under-segment or over-segment. To improve accuracy, more complex loss functions that heavily penalized incorrect boundary predictions were employed, and the resolution of input images was increased to provide more detailed information to the model. Another issue was the disparity in class distribution, as certain categories of anomalies were less frequent, leading to skewed learning. This was remedied by synthetic data augmentation, over-sampling underrepresented classes, and adjusting class weights during loss computation to prioritize learning for rarer anomalies.

(3) Challenges in Model Training

Training the YOLOv8 model for instance segmentation also presented several hurdles. One significant challenge was overfitting, where the model performed well on training images but poorly on validation images. This was mitigated by introducing regularization techniques such as dropout and early stopping, and by expanding data augmentation to include a greater variety of transformations, enhancing the model’s generalization capabilities. The computational intensity of training, particularly given the large sizes of generated images, posed another substantial challenge. This was addressed by transitioning training to cloud-based platforms with more robust GPU support and employing techniques such as model checkpoints and mixed-precision training to optimize computational resources. Finally, selecting appropriate evaluation metrics that accurately reflected the model’s performance across different types of anomalies was challenging. A comprehensive set of metrics including precision, recall, F1-score, and mean average precision (mAP) at various Intersection over Union (IoU) thresholds were utilized to ensure a thorough assessment of the model's effectiveness.

5.5 Summary of Findings

This thesis presents significant findings on the utilization of powerful generative AI and instance segmentation models to detect anomalies in aerial photography. Here is a summary of the main discoveries:

(1) Generative Model Proficiency: The unCLIP model demonstrated a robust ability to generate aerial images from text descriptions. The images, rich in detail and diversity, provided an expansive visual dataset for anomaly detection tasks, underscoring the potential of text-to-image AI models in generating training data where real datasets are scarce or sensitive.

(2) Image Realism and Relevance: The images created were found to be strikingly realistic and contextually relevant to the input prompts. While the overall quality was high, there was variability in the degree of realism, highlighting the intricate balance between generative model capabilities and the specificity of textual prompts.

(3) Human Perception of AI-Generated Imagery: The public survey revealed that the images generated by unCLIP were largely perceived as realistic. Participants were able to discern anomalies with a reasonable degree of accuracy, indicating the viability of these images for practical applications.

(4) Image Quality Metrics: Quantitative assessments such as SSIM and PSNR provided mixed results. While PSNR values were close to acceptable thresholds for image quality, SSIM scores indicated significant room for improvement in capturing the complexity of real-world textures and structures.

(5) YOLOv8 Model's Learning Trajectory: The instance segmentation model, YOLOv8, showed marked improvement in identifying and classifying anomalies over 25 training epochs. The training process revealed the model's ability to learn from a diverse dataset and refine its predictive accuracy over time.

(6) Accuracy Metrics and Model Performance: Evaluation metrics such as precision, recall, and mAP scores improved as the model training progressed, showcasing the model’s increasing adeptness at segmenting and classifying instances within the aerial images. The precision-recall curves from the validation datasets provided evidence of the model's capacity to differentiate between normal and anomalous features within the aerial images effectively.

(7) Challenges and Adaptations: The research faced several challenges, notably in image generation fidelity, segmentation accuracy, and training optimization.

These were addressed through iterative refinements, model adjustments, and computational enhancements, thereby advancing the field's understanding of applying AI in anomaly detection within aerial imagery.

The implications of these findings are multifaceted for the field of aerial imagery analysis and anomaly detection. They demonstrate the promise of using AI to supplement or even substitute real-world data collection in contexts where data may be unavailable or difficult to collect. Moreover, the capacity to train models with synthetic data opens the door to a more flexible and scalable approach to model training, especially in fields like environmental monitoring, urban planning, and disaster response where current events may suddenly create demand for rapid model adaptation.

5.6 Summary of this Chapter

Chapter 5 analyzes aerial images generated using the unCLIP model to assess their realism, relevance, and anomaly clarity. The unCLIP model demonstrated its ability to create aerial visuals from textual prompts, showing characteristics like diversity, detail, and quality. A public survey and image quality metrics (SSIM and PSNR) confirmed the images' realism, quality, and anomaly detection ability. While effective in depicting anomalies, they need improvement for indistinguishable realism.

Metrics like Box Loss, Segmentation Loss, Precision, Recall, and mAP scores over 25 epochs showed the model's learning curve. Although initial epochs had higher losses and lower precision/recall, the model improved steadily, achieving significant progress by the 25th epoch. F1 score and precision-recall curves illustrated the model's improvement.

Key insights emphasized model performance enhancements, optimal confidence thresholds, and visual validation. The study highlighted challenges like image generation, segmentation accuracy, and model training while recognizing AI's potential in anomaly detection. It also acknowledged limitations in dataset quality and generalization, emphasizing the need for further research.

Chapter 6 Conclusion and Future Work

The thesis concludes by reflecting on the exploration into the use of generative AI and instance segmentation models for aerial imagery analysis and anomaly detection. It underscores the transformative potential and existing capabilities that these technologies bring to the field. Through comprehensive research and evaluations, this thesis demonstrates that generative models like unCLIP are capable of producing realistic and high-quality images from textual prompts. These images significantly aid in training and enhancing instance segmentation models like YOLOv8. The results from these models in detecting and categorizing anomalies mark a substantial advancement in applying machine learning to aerial imaging.

The analysis incorporates public surveys and both qualitative and quantitative assessments, which generally regard the generated images as realistic and indicative of the models' advanced learning capabilities. However, challenges remain in capturing the detailed textures and structures that mirror the real world closely. Limitations identified in the representativeness of the dataset and the generalizability of the model highlight the necessity for ongoing improvements in methodology and model sophistication.

Future Work:

Looking ahead, the thesis outlines a proactive plan to enhance the research field. Future work will focus on refining the quality and diversity of the generated images to better train models capable of accurate anomaly detection. Building more extensive and varied datasets will improve the robustness and applicability of the models. Implementing more sophisticated image quality metrics that reflect human perception more accurately is also a priority. Additionally, validating the models with real-world data will ensure they perform effectively outside controlled environments. Emphasizing the importance of ethical implications in AI development, the thesis advocates for norms that govern acceptable practices in imagery generation.

The conclusion reiterates that the end of this thesis represents a milestone rather than a final endpoint in the ongoing journey of research. By continuously refining the technology and methods, the field of aerial imagery and anomaly detection is expected to advance not only technically but also in aligning responsibly with the broader implications of artificial intelligence advancements.

Future work aims to tackle these challenges by exploring cross-domain applications to extend the benefits of anomaly detection models to fields like medical imaging or industrial inspections. Developing dynamic anomaly models that incorporate temporal data and predictive analytics could potentially revolutionize how we understand changes over time. Enhancing the explainability and openness of AI models will also encourage increased user approval and confidence.

Furthermore, the integration of technologies like GIS (Geographic Information Systems) and IoT (Internet of Things) may make real-time monitoring and feedback systems for anomaly identification and quick response conceivable. Drones with artificial intelligence and other self-governing technology can be used to spot and report abnormalities in this sector of the economy.

By addressing these avenues, future research will significantly extend the capabilities of studying aerial imagery, leading to more precise, effective, and ethically sound applications in anomaly detection. This conclusion lays a robust foundation for future investigations, setting the stage for continued exploration that will push the boundaries of aerial imagery analysis while prioritizing responsible and beneficial uses of AI.

References

[1] Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., ... & Sutskever, I. Zero-shot text-to-image generation[J]. International conference on machine learning, 2021: 8821-8831.

[2] Everaerts, J. The use of unmanned aerial vehicles (UAVs) for remote sensing and mapping[J]. The international archives of the photogrammetry, remote sensing and spatial information sciences, 2008: 1187-1192.

[3] Ramesh A, Dhariwal P, Alex N, et al. Hierarchical Text-Conditional Image Generation with CLIP Latents[DB/OL]. (2022-04-13) [2023-12-11]. https://arxiv.org/abs/2204.06125.

[4] Radford A, Kim J W, Hallacy C, et al. Learning Transferable Visual Models from Natural Language Supervision[C]. Proceedings of the 38th International Conference on Machine Learning, Online: PMLR, 2021: 139: 8748-8763.

[5] Pontecorvo, C., & Sherrah, J. Anomaly Detection of Man-Made Objects in Large Aerial Images[J]. 2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA), 2015: 1-8.

[6] Lin, A. Y. -M., Novo, A., Har-Noy, S., Ricklin, N. D., & Stamatiou, K. Combining GeoEye-1 Satellite Remote Sensing, UAV Aerial Imaging, and Geophysical Surveys in Anomaly Detection Applied to Archaeology[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2011, 4(4): 870-876. DOI: 10.1109/JSTARS.2011.2143696.

[7] Tayba, M., & Rivas, P. Enhancing the Resolution of Satellite Imagery Using a Generative Model[J]. 2021 International Conference on Computational Science and Computational Intelligence (CSCI), 2021: 20-25.

[8] Avola, D., Cannistraci, I., Cascio, M., Cinque, L., Diko, A., Fagioli, A., ... & Pannone, D. A novel GAN-based anomaly detection and localization method for aerial video surveillance at low altitude[J]. Remote Sensing, 2022, 14(16): 4110.

[9] Colomina, I., & Molina, P. Unmanned aerial systems for photogrammetry and remote sensing: A review[J]. ISPRS Journal of photogrammetry and remote sensing, 2014, 92: 79-97.

[10] Mooney, P., & Corcoran, P. Characteristics of heavily edited objects in OpenStreetMap[J]. Future Internet, 2012, 4(1): 285-305.

[11] Lary, D. J., Alavi, A. H., Gandomi, A. H., & Walker, A. L. Machine learning in geosciences and remote sensing[J]. Geoscience Frontiers, 2016, 7(1): 3-10.

[12] Gonzalez, R. C. Digital image processing[M]. Pearson education India, 2009.

[13] James, G., Witten, D., Hastie, T., & Tibshirani, R. An introduction to statistical learning[M]. New York: Springer, 2013, Vol. 112: 18.

[14] Liu, X., Ghazali, K.H., Han, F., & Mohamed, I.I. Review of CNN in aerial image processing[J]. The Imaging Science Journal, 2023, 71(1): 1-13.

[15] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. Generative adversarial networks[J]. Communications of the ACM, 2020, 63(11): 139-144.

[16] Kingma, D. P., & Welling, M. Auto-encoding variational bayes[DB/OL]. arXiv preprint arXiv:1312.6114, 2013.

[17] Karras, T., Aila, T., Laine, S., & Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation[DB/OL]. arXiv preprint arXiv:1710.10196, 2017.

[18] Zhang, H., Goodfellow, I., Metaxas, D., & Odena, A. Self-attention generative adversarial networks[C]. International conference on machine learning, 2019: 7354-7363.

[19] Elgammal, A., Liu, B., Elhoseiny, M., & Mazzone, M. Can: Creative adversarial networks, generating" art" by learning about styles and deviating from style norms[DB/OL]. arXiv preprint arXiv:1706.07068, 2017.

[20] Ye, H., Zhang, J., Liu, S., Han, X., & Yang, W. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models[DB/OL]. arXiv preprint arXiv:2308.06721, 2023.

[21] Purushwalkam, S., Gokul, A., Joty, S., & Naik, N. BootPIG: Bootstrapping Zero-shot Personalized Image Generation Capabilities in Pretrained Diffusion Models[DB/OL]. arXiv preprint arXiv:2401.13974, 2024.

[22] Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., & Lee, H. Generative adversarial text to image synthesis[C]. International conference on machine learning, 2016: 1060-1069.

[23] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation[DB/OL]. arXiv preprint arXiv:1406.1078, 2014.

[24] Contreras-Cruz, M.A., Correa-Tome, F.E., Lopez-Padilla, R., & Ramirez-Paredes, J.P. Generative Adversarial Networks for anomaly detection in aerial images[J]. Computers and Electrical Engineering, 2023, 106: 108470.

[25] Isola, P., Zhu, J., Zhou, T., & Efros, A. A. Image-to-Image Translation with Conditional Adversarial Networks[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017: 1125-1134.

[26] Denton, E., Chintala, S., Szlam, A., & Fergus, R. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks[C]. Advances in Neural Information Processing Systems (NeurIPS), 2015: 1486-1494.

[27] Ariuntuya, A. How to Train YOLOv8 Instance Segmentation on a Custom Dataset[DB/OL]. Roboflow Blog, 2023-05-26. https://blog.roboflow.com/how-to-train-yolov8-instance-segmentation/

[28] Ding, Y., Tian, C., Ding, H., & Liu, L. The CLIP Model is Secretly an Image-to-Prompt Converter[C]. Advances in Neural Information Processing Systems, 2024, 36.

[29] Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?[C]. Proceedings of FAccT, 2021.

[30] Lipton, Z. C. The Mythos of Model Interpretability[J]. Queue, 2018, 16(3): 31-57.

[31] Dosovitskiy, A., et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale[DB/OL]. arXiv preprint arXiv:2010.11929, 2020.

[32] Vaswani, A., et al. Attention is All You Need[C]. Advances in Neural Information Processing Systems, 2017, 30.

[33] He, K., et al. Deep Residual Learning for Image Recognition[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: 770-778.

[34] Liu, Z., et al. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows[DB/OL]. arXiv preprint arXiv:2103.14030, 2021.

[35] Srivastava, N., et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting[J]. Journal of Machine Learning Research, 2014, 15(1): 1929-1958.

[36] Krogh, A., & Hertz, J. A. A Simple Weight Decay Can Improve Generalization[C]. Advances in Neural Information Processing Systems, 1992, 4.

[37] Smith, L. N. Cyclical Learning Rates for Training Neural Networks[C]. IEEE Winter Conference on Applications of Computer Vision (WACV), 2017: 464-472.

[38] Tan, M., & Le, Q. V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks[C]. Proceedings of the 36th International Conference on Machine Learning (ICML), 2019.

[39] Finn, C., Abbeel, P., & Levine, S. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks[C]. Proceedings of the 34th International Conference on Machine Learning (ICML), 2017.

[40] Shin, H. C., et al. Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning[J]. IEEE Transactions on Medical Imaging, 2016, 35(5): 1285-1298.

[41]斋藤康毅. 深度学习入门: 基于 Python 的理论与实现[M]. 陆宇杰, 译. 北京: 人民邮电出版社, 2018.

[42] 王鹏雄. 基于生成对抗网络的文字到图像的生成研究[D]. 重庆理工大学, 2023.

[43] 彭斌，白静，李文静，等.面向图像分类的视觉 Transformer 研究进展[J].计算机科学与探索，2024，18(02):320-344.

[44] 钟淑瑛，李陶深，张敏.一种基于 PCA 技术的入侵检测特征提取方法[C]//广西计算机学会.广西计算机学会 2005 年学术年会论文集.广西大学计算机与电子信息学院;广西大学计算机与电子信息学院;广西大学计算机与电子信息学院;，2005:3.

[45] 陈子民，关志涛.基于条件扩散模型的图像分类对抗样本防御方法[J/OL].计算机工程，1-11[2024-05-17].

Acknowledgments

Without the invaluable assistance, motivation, and steadfast backing of numerous individuals, this research study would not have been possible.

Mr. Li Yun Peng, my supervisor, selected me to be a part of his research team, and I am very grateful to him. Thanks to his expertise, insightful criticism, unwavering support, and inspiration, I was able to raise the bar on this thesis document. His invaluable feedback and support shaped the course of this thesis and helped me persevere through challenging times.

Next, I would like to honor my university: Tiangong University, both the departments that I belong Tiangong International and the School of Artificial Intelligence and all the teachers who have taught me. Tiangong University is the place where my passion became a possibility. Tiangong University is the only place on this planet where I became confident and proud of who I am. I love this place dearly.

Then, I would pay tribute to my friends who supported me with this thesis study - Chanduni Nethmini, Ranadi Tuala, Bin Wen, Yuni Dewi, and my classmates, study mates who had helped me whenever I was in need. Their friendship and fruitful debates have given the severe concentration needed for this research the much-needed counterbalance.

A warm hearty thanks to my family! I am so grateful to my family for their consistent love, support and patience, which have given me the drive to pursue my interests. Thank you for not giving up on me. My academic journey has been built upon the basis of your love, trust and encouragement.

Finally, I would like to thank OpenAI for creating the state-of-the-art models like CLIP and unCLIP that have allowed for the completion of this study. Their inventiveness and commitment to the progress of artificial intelligence have created new opportunities for investigation and learning in the field of aerial imagery.

Thank you all for your invaluable support.

Nethmi Muthugala (那娜)