Chair of Photogrammetry and Remote Sensing
TUM School of Engineering and Design
Technical University of Munich

Geospatial-Linguistic Alignment for Multimodal Satellite Scene Description

Scientific work to obtain the degree

Master of Science (M.Sc.)

at the TUM School of Engineering and Design
of the Technical University of Munich.

Supervised by Prof. Dr.-Ing. Michael Eineder
Chair of Photogrammetry and Remote Sensing

Dr. Daniela Espinoza Molina
Dr. Luca Chiarabini
German Aerospace Center (DLR), Remote Sensing Technology Institute

Submitted by Clara Dorothea Väth (03783156)

Submitted on September 30, 2026

Acknowledgments

First, I would like to thank Prof. Dr. Michael Eineder for giving me the opportunity to pursue this thesis at DLR and for his supervision.
I am equally grateful to my supervisors, Dr. Daniela Espinoza Molina and Dr. Luca Chiarabini, for their technical guidance, the regular progress meetings that helped keep the project on track, and, above all, for the understanding and humanity they showed me during some life-changing events throughout this work. I am very grateful for the trust and support they gave me during a time when finishing this thesis was sometimes not the only thing that mattered.
I want to thank my parents. My mother, Dr. med. Kunigunda Thaler-Väth, has once again shown me what a person can overcome, however big the task may seem. Her strength has been an important source of motivation throughout the final months of this thesis and has reminded me many times to keep going, even when things seemed difficult. My father, Rainer Väth, has been the constant who held everything together throughout my whole life, but especially during the past half year. I could not have asked for a more reliable source of support, and I am deeply grateful for everything he has done for me.
I would also like to thank my brother, Benedict, who lives out discipline on a different level and has taught me that even unpleasant tasks are never as overwhelming as they seem once you simply start them. This lesson has been useful more often than he probably realises.
To my friends from the master's programme, Arianna, Ioanna, Vangelis, Antonio, and Ges, thank you for your enthusiasm for our field of study, which showed itself in the several projects we worked on together. You made the past years so much more enjoyable, and I am very grateful for all the conversations, the shared struggles, and the good times along the way. I am already looking forward to many more adventures together.
Finally, I want to thank Niclas, who was writing his own thesis at the same time. Thank you for your patience, for the sleepless nights of writing that we shared, and for the countless technical discussions whenever one of us got stuck. Having someone beside me who understood exactly what this period felt like made the difficult parts much easier. I am especially grateful that, even when both of us were tired and stressed, we could still find ways to make each other laugh.

Abstract

Satellite missions such as Sentinel-1 and Sentinel-2 produce increasingly large amounts of data, which makes manual analysis difficult in many cases. Vision-language models (VLMs) can help interpret this data by describing satellite scenes in natural language. Many Earth-observation VLMs, however, use encoders pretrained on natural RGB images, obtain SAR supervision from optical descriptions, or process radar and optical features together. It is therefore difficult to see how much a generated description is based on information from an individual sensor. This is especially relevant for Sentinel-1, which can provide observations under cloud cover and at night. In these situations, the description has to rely on SAR information alone.

This thesis examines whether a frozen geospatial foundation model can provide the visual input for scene descriptions from Sentinel-1 and Sentinel-2, and how strongly the generated captions depend on the observed scene. The frozen TerraMind encoder is connected to the frozen LLaVA-1.5-7B language model through a small trainable bridge. Two bridge variants are tested, a linear projection and a two-layer multilayer perceptron (MLP). Using the SSL4EO-S12 v1.1 dataset, Sentinel-1 and Sentinel-2 are treated as separate tracks, each with its own bridge and supervision. Both caption datasets were generated automatically by language models. For Sentinel-1, a new dataset was created by prompting a VLM to describe a false-colour composite. The SAR supervision therefore does not depend on optical imagery.

The captions are evaluated on 5,120 test image–caption pairs using BLEU-4, ROUGE-L, METEOR and SBERT cosine similarity. Two additional controls are used to check whether the model uses the visual input. One compares the generated captions with reference captions from other scenes, while the other uses a retrieval evaluation. The original LLaVA-1.5 serves as a zero-shot baseline with its vision encoder pretrained on natural images. All trained conditions achieve higher scores than the baseline on all four metrics. The correct scene also consistently results in higher scores than a mismatched reference. The retrieval results are different for the two tracks. For Sentinel-1, the trained bridge ranks the correct scene considerably higher than the baseline, while for Sentinel-2, the baseline performs similarly to or slightly better than the trained bridge. Changing the bridge non-linearity or the decoder has only a small effect on the results.

Overall, the results show that a small trainable bridge can generate fluent descriptions from a frozen geospatial encoder that have a measurable connection to the input scene, including for SAR alone. Since changing the bridge and decoder has only a small effect, the remaining limitations may be related to the encoder representation or the caption supervision, neither of which was changed in this study. The trained model also largely reproduces the regular phrasing of the reference captions, which is partly introduced by the prompts used to generate the datasets.

Contents

List of Figures **VII**

List of Tables **IX**

List of Abbreviations **XI**

1 Introduction **1**
1.1 Motivation . . . 1
1.2 Thesis Contribution . . . 2
1.3 Outline of the Thesis . . . 2

2 Foundations and Related Work **5**
2.1 Deep Learning Architectures . . . 5
2.1.1 Multilayer Perceptron (MLP) . . . 5
2.1.2 Activation Function . . . 6
2.1.3 Transformer Encoder . . . 6
2.1.4 Vision Transformer . . . 7
2.2 Foundation Models . . . 7
2.2.1 Contrastive Pretraining, intra-modal . . . 8
2.2.2 Contrastive Pretraining, inter-modal: CLIP . . . 8
2.2.3 Reconstructive and Masked Pretraining . . . 8
2.2.4 Extensions to Earth Observation . . . 8
2.3 Large Language Models . . . 9
2.3.1 Autoregressive language modelling . . . 9
2.3.2 Generative pretraining and transfer . . . 9
2.3.3 Scaling and in-context learning . . . 9
2.3.4 The LLaMA family . . . 10
2.4 Vision Language Models . . . 10
2.5 Geospatial Foundation Models . . . 11
2.5.1 Language-free multi-sensor Foundation Models . . . 12
2.5.2 Contrastive FM adaptions to Remote Sensing . . . 12
2.5.3 Cross-modal reconstruction: TerraMind . . . 13
2.5.4 Remote Sensing VLMs . . . 14
2.6 Research Gap . . . 15

3 Data **17**
3.1 Sentinel-1 and Sentinel-2 . . . 17
3.1.1 The Copernicus open data policy . . . 17
3.1.2 Sentinel-2: passive multispectral imaging . . . 17
3.1.3 Sentinel-1: active microwave imaging . . . 18
3.1.4 Complementarity and cloud dependency . . . 19
3.2 SSL4EO-S12 v1.1 . . . 20
3.3 Automated Caption Generation for SSL4EO-S12 . . . 23
3.3.1 Llama3-SSL4EO-S12 . . . 23
3.3.2 SAR captions . . . 25
3.4 Data Handling . . . 27

4 Methodology **31**
4.1 Overview . . . 31
4.2 TerraMind Encoder . . . 33
4.2.1 Checkpoint loading and freezing . . . 33
4.2.2 Input resolution and token layout . . . 33
4.2.3 Modality handling . . . 34
4.2.4 Forward pass . . . 34
4.3 LLM Decoder . . . 35
4.3.1 Checkpoint loading and freezing . . . 35
4.3.2 Prompt template and prompt-embedding cache . . . 35
4.3.3 Training objective and sequence assembly . . . 36
4.3.4 Inference . . . 36
4.3.5 Alternative LLM . . . 37
4.4 Bridge . . . 37
4.4.1 Role and shape . . . 37
4.4.2 Linear projection . . . 39
4.4.3 MLP projection . . . 39
4.4.4 Implementation and number of parameters . . . 40
4.5 Training Objective and Protocol . . . 40
4.5.1 Loss . . . 41
4.5.2 Optimizer . . . 41
4.5.3 Learning-rate schedule . . . 41
4.5.4 Diagnostic measures . . . 42
4.6 Design Choices . . . 42
4.6.1 Encoder choice . . . 43
4.6.2 Bridge-only training without LoRA . . . 43

5 Experiments **45**
5.1 Experimental Setup . . . 45
5.1.1 Data and Evaluation Splits . . . 45

5.1.2 Training Protocol and Fixed Configuration 46
5.1.3 Selection of the Learning Rate 47
5.1.4 Reproducibility 48
5.2 Conditions under Comparison 48
5.2.1 Design Factors 49
5.2.2 Control Condition 50
5.2.3 External Baseline 51
5.2.4 Summary of Conditions 52
5.3 Evaluation Metrics 52
5.3.1 Definition of the Metrics 53
5.3.2 Implementation and Reported Quantities 54
5.3.3 Retrieval Measures 55
5.4 Optical Captioning Results 57
5.4.1 Training Behaviour 57
5.4.2 Generated Captions 58
5.4.3 Metric Results 59
5.5 SAR Captioning Results 60
5.5.1 Training Behaviour 60
5.5.2 Generated Captions 61
5.5.3 Metric Results 63
5.6 Contribution of the visual input to performance 65
5.6.1 Performance without the Correct Scene 65
5.6.2 Retrieval-based Check 66
5.7 Discussion of the Results 68
5.7.1 Comparison of Training Conditions 68
5.7.2 Contribution of the Bridge Training 69
5.7.3 Strengths and Weaknesses of the Generated Captions 71

6 Conclusion **73**
6.1 Summary 73
6.2 Future Work 74
6.2.1 Broader and task-oriented datasets 74
6.2.2 Instruction tuning and parameter-efficient adaptation of the decoder 75
6.2.3 A language-aligned SAR encoder as a point of comparison . . . 76
6.2.4 Joint Sentinel-1 and Sentinel-2 input 76
6.2.5 Using TerraMind's generative capabilities 77

Bibliography **79**

A Prompting **89**
A.1 SAR-text Dataset Creation 89
A.2 SAR-specific Prompt for Decoder 90

A.3 Prompts for Baseline 91

B Training Details **93**
B.1 Fine-tuning 93
B.2 Configuration files 93
B.3 Learning-rate sweeps 94
B.4 Grounding gap 95

C Computation **99**
C.1 Hardware and software 99
C.2 Computational Cost 99
C.2.1 Training Cost 99
C.2.2 Inference Cost 100

D Additional Examples of the Results **103**

List of Figures

3.1 Spectral reflectance signature examples. 18
3.2 Idealised scattering mechanisms determining the backscatter coefficient. 19
3.3 Spatial distribution of the SSL4EO-S12 dataset samples globally. . . 21
3.4 Examples of the SSL4EO-S12 dataset. 22
3.5 Examples of the Llama3-SSL4EO-S12 captions. 24
3.6 Examples of the Gemini-generated S1 captions. 26
3.7 Dataset directory structure. 28

4.1 Alignment Pipeline and Training Loop 32

5.1 Artefacts of a training run and a hyperparameter sweep. 49
5.2 Training and validation loss per epoch for the winner trial of the optical track. 57
5.3 Examples of the generated S2 captions. 58
5.4 Training and validation loss per epoch for the winner trial of the SAR track. 61
5.5 Examples of the generated S1 captions. 62

B.1 Manual stage on Sentinel-2 with the MLP bridge at three learning rates. 94
B.2 Validation loss of the winning trial of each study. 96

D.1 Scene 35129, acquisition time 2. 104
D.2 Scene 231325, acquisition time 3. 105
D.3 Scene 20193, acquisition time 3. 106
D.4 Scene 145981, acquisition time 0. 107
D.5 Scene 224582, acquisition time 0. 108
D.6 Scene 111330, acquisition time 3. 109
D.7 Scene 94762, acquisition time 3. 110

List of Tables

3.1 Summary of modalities included in SSL4EO-S12 v1.1 and used in this work. 21

4.1 Trainable tensors of the bridge variants. 40

5.1 Constant configuration values for all runs. 46
5.2 Experiment Design factors. 50
5.3 Summary of Evaluation Conditions. 52
5.4 Evaluation results of the optical track. 59
5.5 Evaluation results of the SAR track. 63
5.6 Scores compared to shifted-captions. 65
5.7 Retrieval of the matching reference among all the references of the dataset. 67

B.1 Sweep conditions. 95
B.2 Winning trials. 95

C.1 Computational cost of the seven learning-rate sweeps 100
C.2 Computational cost of the inference. 101

List of Abbreviations

AdamW	Adaptive Moment Estimation with Decoupled Weight Decay
AI	Artificial Intelligence
API	Application Programming Interface
BERT	Bidirectional Encoder Representations from Transformers
BLEU	Bilingual Evaluation Understudy
CC BY	Creative Commons Attribution
CLIP	Contrastive Language-Image Pretraining
CLS	Classification (Token)
CNN	Convolutional Neural Network
C-SAR	C-Band Synthetic Aperture Radar
CSV	Comma-Separated Values
CUDA	Compute Unified Device Architecture
dB	Decibel
DEM	Digital Elevation Model
DLR	Deutsches Zentrum für Luft- und Raumfahrt (German Aerospace Center)
DN	Digital Number
EO	Earth Observation
EOS	End of Sequence
ESRI	Environmental Systems Research Institute
FFN	Feed-Forward Network
FM	Foundation Model
GEE	Google Earth Engine
GELU	Gaussian Error Linear Unit
GeoFM	Geospatial Foundation Model
GPT	Generative Pretrained Transformer
GPU	Graphics Processing Unit
GRD	Ground Range Detected
HPDA	High-Performance Data Analytics
ID	Identifier
IW	Interferometric Wide Swath
L1C	Level-1C
L2A	Level-2A
LLaMA	Large Language Model Meta AI
LLaVA	Large Language and Vision Assistant
LLM	Large Language Model
LMM	Large Multimodal Model

LN	Layer Normalisation
LoRA	Low-Rank Adaptation
LRZ	Leibniz-Rechenzentrum (Leibniz Supercomputing Centre)
LULC	Land Use and Land Cover
MAE	Masked Autoencoder
METEOR	Metric for Evaluation of Translation with Explicit Ordering
MLLM	Multimodal Large Language Model
MLP	Multilayer Perceptron
MoCo	Momentum Contrast
NDVI	Normalised Difference Vegetation Index
NIR	Near Infrared
NLTK	Natural Language Toolkit
Q-Former	Querying Transformer
R@k	Recall at Position k
ReLU	Rectified Linear Unit
RGB	Red, Green, Blue
ROUGE-L	Recall-Oriented Understudy for Gisting Evaluation, Longest Common Subsequence
S1	Sentinel-1
S2	Sentinel-2
SAR	Synthetic Aperture Radar
SBERT	Sentence-BERT
SSL4EO-S12	Self-Supervised Learning for Earth Observation, Sentinel-1/2
VH	Vertical transmit, Horizontal receive
ViT	Vision Transformer
VLM	Vision-Language Model
VV	Vertical transmit, Vertical receive
YAML	YAML Ain't Markup Language

1 Introduction

1.1 Motivation

The amount of high-resolution satellite data from missions such as Sentinel-1 and Sentinel-2 is increasing continuously and already exceeds the limit to be analysed manually in many cases. This creates a need for automated methods that can support the semantic interpretation of satellite imagery.

The development of Vision-Language Models (VLMs) has made it possible to work with visual information using natural language. In Earth observation (EO), this offers new ways to interact with satellite imagery. Instead of developing a separate classifier for each task, an analyst could describe a scene, ask about its land cover, or ask for an explanation of an observed change. This can be useful in applications where a fast and flexible analysis is needed, for example in disaster assessment, agricultural monitoring, or humanitarian intelligence.

Although general-purpose VLMs have improved considerably, their use in remote sensing is still challenging. Many existing EO-VLMs are based on architectures that were originally developed for natural RGB images and are adapted to Sentinel-2 optical data through additional training. Remote-sensing data, however, provide different types of information. Multispectral optical imagery contains information beyond the visible spectrum, while SAR measures microwave backscatter instead of visual appearance. The two modalities also have different acquisition conditions and provide complementary information. Sentinel-2 is affected by cloud cover and requires daylight, while Sentinel-1 can acquire observations independently of these conditions. SAR is therefore particularly useful for monitoring in areas with frequent cloud cover. Existing multimodal EO-VLMs process SAR and optical features together in one pathway. As a result, it is difficult to determine how much of the generated language is based on the information from each sensor. A system for language-based interpretation needs representations that reflect the characteristics of the individual sensing modalities and a way to transfer this information to the language model.

The information that the language model can use depends not only on the EO encoder but also on the bridge that transfers the encoder representations into the language space. Information from the encoder can be changed or lost during this step. If the encoder, bridge, and language model are all trained or adapted at the same time, it is difficult to determine which component is responsible for an improvement in performance. In this thesis, the encoder and language model are kept frozen and only the bridge is trained. This allows the effect of the alignment stage to be studied more

directly. At the same time, limitations in the generated captions can give an indication of the information available in the encoder representation and the supervision used during training.

1.2 Thesis Contribution

The main contributions of this thesis are:

1. A minimal bridge architecture with a linear and multilayer perceptron (MLP) variation connecting a frozen multimodal remote-sensing encoder to a frozen large language model, with only the projection module trained. The pipelines are deliberately kept separate to isolate the modality-specific contribution to language alignment.
2. A Sentinel-1 caption dataset in which the linguistic supervision is generated directly from the SAR observation using a vision-language model applied to a VH/VV/VV–VH false-colour composite.
3. An end-to-end captioning pipeline from Sentinel-1 and Sentinel-2 data to natural language image descriptions.
4. An evaluation protocol that combines single-factor comparisons with grounding controls. Each comparison changes only one component to examine the effect of non-linearity in the bridge, the visual instruction tuning of the decoder, or a sensor-aware prompt. The shifted-captions condition and the retrieval evaluation with a defined chance level are then used to assess whether the generated captions depend on the correct scene or mainly reflect the style of the reference corpus.

The training setup in this thesis is intentionally kept minimal. Only the bridge, with around 3 million, respectively 20 million parameters, is trained, while both the remote-sensing encoder and the large language model remain frozen. This differs from most remote-sensing captioning approaches, which also fine-tune the language model, either fully, selectively, or with parameter-efficient methods. Training only the bridge makes it possible to isolate the effect of the alignment step and to evaluate how much captioning performance can be achieved without changing the language model itself.

1.3 Outline of the Thesis

The thesis is structured into five main chapters, which cover the theoretical background, the data and methodology, the experiments, and the conclusions.

Chapter 2 provides the theoretical background on vision-language models and geospatial foundation models, with particular focus on the representations and alignment methods relevant to this work. Existing remote-sensing vision-language models are also reviewed, leading to the research gap addressed in this thesis. The datasets and captioning procedures used for training and evaluation are described in Chapter 3. In Chapter 4, the proposed architecture and training procedure are presented, including the TerraMind encoder, the bridge, and the language model. The resulting models are evaluated in Chapter 5, where the Sentinel-1 and Sentinel-2 tracks are examined under the experimental settings defined in the methodology. Finally, Chapter 6 discusses the limitations of the approach and presents possible directions for future research.

2 Foundations and Related Work

2.1 Deep Learning Architectures

Deep learning refers to neural networks combining multiple layers of operations that learn feature representations directly from the input data instead of relying on handcrafted features [1]. This section introduces the architectural building blocks from which the encoder, the bridge, and the decoder of this work are constructed.

2.1.1 Multilayer Perceptron (MLP)

Goodfellow et al. regard the feedforward deep network, also called MLP, as “the quintessential example of a deep learning model” [2, p. 5]. Such a network approximates a relationship between inputs and outputs through a sequence of simpler functions. Instead of computing the mapping in a single step, the network transforms the input over several stages, with each stage passing an intermediate representation to the next. [2, p. 5] An MLP is composed of a sequence of affine transformations of the form $z = Wa + b$, where W is the weight matrix, b is the bias vector, and a is the input vector from the previous layer. With an elementwise nonlinear activation function σ applied between the layers, it results in $a_l = \sigma(Wa_{l-1} + b)$. An L-layer network can be written as

$$a_L = W_L\,\sigma\big(W_{L-1}\,\sigma(\dots\sigma(W_1 a_0 + b_1)\dots) + b_{L-1}\big) + b_L. \tag{2.1}$$

a_0 denotes the input vector, l the index of an intermediate layer and L the index of the output layer. [1]

The internal parameters are adjusted by *back-propagation*, which LeCun et al. [1] describe as "a practical application of the chain rule for derivatives," applied repeatedly to propagate gradients from the output layer down to the input. In practice, these gradients are used by *stochastic gradient descent*. For each training step, a small batch of examples is used to calculate the corresponding outputs and errors. The gradients obtained from these examples are then averaged and used to adjust the model parameters. [1]

2.1.2 Activation Function

If σ is removed from Equation (2.1), the big affine transformations of the different layers can be combined into one affine transformation and the network would still describe a linear relationship between the input and the output, even if more layers were added. The depth of the network would then not lead to a stronger ability to represent more complex functions [2, p. 171].

The *rectified linear unit (ReLU)*,

$$\mathrm{ReLU}(x) = \max(0, x), \tag{2.2}$$

is widely used as a standard choice for activation functions in feedforward neural networks [2, p. 175]. It appears for example in the feed-forward block of the original Transformer, $\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$ [3].

Hendrycks and Gimpel [4] propose the *Gaussian error linear unit* as an alternative,

$$\mathrm{GELU}(x) = x \cdot \Phi(x), \tag{2.3}$$

where Φ represents the cumulative distribution function of the standard normal distribution. Unlike ReLU, which determines whether an input is passed through based on its sign, GELU applies a continuous scaling that depends on the input value. The authors describe GELU as a smooth approximation of ReLU, with both functions approaching the same behaviour for large input values. They also report that GELU has become a common default choice in state-of-the-art Transformer architectures, including BERT and GPT [4].

2.1.3 Transformer Encoder

Transformers rely entirely on attention mechanisms and therefore do not require recurrence or convolution, which were predominant in models before. Its main operation, *scaled dot-product attention*, compares how well a query and a set of key-value pairs match and based on that produces a weighted combination of the corresponding values [3]:

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V. \tag{2.4}$$

The term *self-attention* is used when the queries, keys, and values are generated from one common sequence. In this setup, every position in the sequence can access information from all other positions within a single layer. As a result, relationships between positions can be represented without increasing the number of sequential operations with sequence length, and the maximum path length remains $O(1)$. A

recurrent layer needs $O(n)$ sequential operations to establish such connections and a convolutional layer a maximum path length of $O(\log_k(n))$, where n is the sequence length and k is the kernel size of the convolution. The self-attention mechanism can be further extended through h parallel attention heads, each operating on a learned projection of the input, what enables the model to capture relationships across different representation subspaces at the same time. An encoder layer has a multi-head self-attention sub-layer followed by a position-wise feed-forward network. Since self-attention itself does not encode the order of the input tokens, positional information is added to the token representations through positional encodings. [3]

2.1.4 Vision Transformer

The *Vision Transformer (ViT)* [5] applies the Transformer architecture to images by dividing the input into non-overlapping patches of size P × P, flattening them and treating them as a sequence of $N = HW/P^2$ tokens, which are then linearly projected into a common D-dimensional representation. A learnable classification token is added at the beginning of the sequence, and its final representation is used as the image-level representation. Similar to the original Transformer, one-dimensional positional embeddings are added to the patch tokens to preserve their spatial order. The resulting sequence is processed by Transformer blocks following the architecture described above, including GELU activations in the feed-forward networks. Whereas CNNs embed locality into their architecture, the ViT has to learn spatial relationships between patches from the training data and the actual image content. [5]

2.2 Foundation Models

The term *Foundation Model (FM)* describes models that are pretrained on broad datasets, typically through large-scale self-supervision, and can subsequently be adapted to a variety of downstream tasks. Typical examples are BERT, GPT-3, and CLIP. Two properties of this approach are especially important: *emergence*, where useful capabilities can develop during training without being explicitly designed, and *homogenization*, where similar machine learning approaches can be applied across different application areas. [6]

Foundation models can broadly be distinguished by whether they learn through contrastive objectives or through reconstructive and masked objectives [7]. This distinction is particularly relevant for the CLIP-based encoders discussed later and for TerraMind.

2.2.1 Contrastive Pretraining, intra-modal

Contrastive learning forms positive relationships between pairs of examples. Samples that are considered similar are rearranged to nearby representations, while dissimilar samples are assigned more distant representations. In SimCLR, two randomly augmented versions of the same image form a positive pair. A shared encoder and projection head are then trained to increase their agreement using a contrastive loss [8]. Momentum Contrast (MoCo) uses a queue of past representations as its source of negative examples and a momentum-updated encoder [9]. In both approaches, the contrastive objective is defined within a single modality.

2.2.2 Contrastive Pretraining, inter-modal: CLIP

CLIP extends the contrastive approach to two different modalities by using paired images and text. During pretraining, the model learns to identify the caption that corresponds to a given image from a large collection of image-text pairs. The visual and textual encoders are trained to map their inputs into a shared embedding space, allowing language to represent visual concepts learned during pretraining. This enables the resulting model to be applied to new tasks without task-specific training. [10] This connection between visual representations and language is important for the VLM and geospatial foundation model approaches discussed in Sections 2.4 and 2.5. However, the approach depends on the availability of large-scale paired image–text datasets, which are not naturally available for Sentinel-1 SAR imagery.

2.2.3 Reconstructive and Masked Pretraining

The basic idea of reconstructive models is to hide a part of the input and train the model to recover the missing information from the remaining context. In the language domain, BERT applies this principle by randomly masking a proportion of the input tokens and predicting their original values from the surrounding context [11]. A similar concept is applied to images by the Masked Autoencoder (MAE), which removes randomly selected image patches and trains the model to reconstruct the corresponding pixels. The high masking ratio of e.g. 75 % used by MAE is possible because images contain substantial spatial redundancy, meaning that missing regions can often be inferred from neighbouring patches. [12]

2.2.4 Extensions to Earth Observation

The reconstructive approach has also been adapted to Earth observation data. SatMAE [13] extends the Masked Autoencoder framework to temporal and multispectral

satellite imagery. The model also uses additional embeddings to represent the acquisition time of each observation, masks image patches independently across different time steps, and represents multispectral channels in groups using separate spectral positional encodings. A foundation model specifically designed for spectral remote sensing imagery is SpectralGPT [14]. Their approach uses a 3D generative Transformer to model the spatial and spectral dimensions jointly and applies multi-target reconstruction to capture relationships between neighbouring spectral bands. Both approaches remain within a single imaging modality, and extensions to multi-sensor and language-aware settings are discussed in Section 2.5.

2.3 Large Language Models

2.3.1 Autoregressive language modelling

An autoregressive language model predicts each token based on the tokens that precede it, modelling the probability of the sequence step by step. The model is trained on unlabelled text by maximising the likelihood of the observed sequences using stochastic gradient descent. Modern language models typically achieve this with a stack of Transformer decoder blocks with causal self-attention. In their description of GPT, the authors explain that the model consists of several Transformer decoder layers. First, multi-head self-attention processes the available context and then position-wise feed-forward layers use this information to predict the probability distribution of the next tokens. [15] The large language models considered below follow this decoder-only architecture, where the attention mechanism is causally masked.

2.3.2 Generative pretraining and transfer

In the GPT paper [15], they introduce a two-stage approach in which a language model is first pretrained on a large and diverse collection of unlabelled text and is then adapted to individual tasks. During this second stage, task-specific input representations are used so that the pretrained model can be transferred with only minor changes to its original architecture. Pretraining and task adaptation can be treated as separate steps.

2.3.3 Scaling and in-context learning

The extent to which the model is updated during adaption is a design choice of the downstream system. It ranges from full fine-tuning to parameter-efficient methods like LoRA [16], to keeping the model entirely frozen, as in in-context learning. The scalability of the autoregressive Transformer architecture was demonstrated with

GPT-3, a model of 175 billion parameters. They evaluated its ability to perform tasks using only a small number of examples provided in the input, without updating the model parameters or applying task-specific fine-tuning. The results showed that larger models improve not only in the zero-shot setting but particularly strongly in the few-shot setting, indicating a growing ability to learn from the given context [17]. Consequently, a sufficiently large pretrained language model can be applied to new tasks by providing appropriate information in the input context. Additionally, downstream use does not necessarily require changes to the model parameters, making fine-tuning only one possible form of adaptation.

2.3.4 The LLaMA family

The LLaMA family is used as the language backbone in many of the vision–language systems discussed in the next sections. LLaMA is a family of foundation language models with model sizes ranging from 7B to 65B parameters, trained on large-scale text data collected from publicly available sources [18]. Llama2 later extended this family to models between 7B and 70B parameters and introduced dialogue-oriented Llama2-Chat variants that were further optimized for conversational tasks [19]. A widely used instruction-tuned model derived from LLaMA is Vicuna, which was developed by fine-tuning LLaMA on user-shared conversations from ShareGPT to improve its ability to follow instructions and engage in dialogue [20]. It is relevant for this work because LLaVA-1.5 uses Vicuna v1.5 as its base language model [21]. The language component of the multimodal system used in Chapter 4 is therefore based on a 7B-parameter LLaMA model that has been further tuned for instruction following and dialogue.

2.4 Vision Language Models

Throughout this thesis, the term *vision–language model (VLM)* is used as a general designation for models that jointly process visual information and natural-language input within a shared or coordinated representation space. This definition includes both discriminative applications, such as image-text retrieval and zero-shot classification, and generative tasks, including image captioning and visual question answering. In the literature, generative models of this type are also referred to as *large multimodal models (LMMs)* [21] or *multimodal large language models (MLLMs)* [22].

The *contrastive dual-encoder* family is represented by CLIP, whose pretraining objective and shared embedding space were introduced in Section 2.2. Its primary applications include image-text retrieval and zero-shot classification, neither of which requires language generation.

Generative vision–language models with a projector bridge combine a pretrained vision encoder, often derived from CLIP, with an autoregressive language model through a comparatively small trainable connector. LLaVA follows this approach by linking a frozen CLIP ViT-L/14 encoder to the Vicuna language model using a linear projection. This projection transforms the visual features into representations that can be processed within the embedding space of the language model [23]. LLaVA-1.5 replaces the linear projection with a two-layer MLP and achieves improved multimodal performance, which demonstrates the benefit of providing the connector with additional capacity [21]. BLIP-2 adopts a similar overall architecture but uses a different form of visual-language connection. Its *Querying Transformer (Q-Former)* uses trainable query representations to extract and compress relevant information from a frozen image encoder before transferring the resulting representation to the language model [24].

Despite these architectural differences, the connector remains relatively small in all of these systems. Depending on the architecture, it consists of a linear projection, a two-layer MLP, or a Q-Former with a parameter count ranging from a few million to several hundred million [21, 24]. Such a connector is unlikely to establish a complete correspondence between arbitrary visual concepts and linguistic representations on its own. Instead, the vision encoder must already provide features that contain a substantial degree of language-compatible semantic information. For natural imagery, this condition is fulfilled by the contrastive pretraining of CLIP described in Section 2.2. Through large-scale image-text training, CLIP learns visual representations that are closely associated with linguistic concepts and can therefore be connected to a language model with a relatively lightweight projection module [10].

2.5 Geospatial Foundation Models

Earth-observation data differ substantially from the natural photographs used in computer vision in terms of sensor characteristics, physical measurement principles, and the object categories they represent. Moreover, satellite imagery does not provide paired image-text datasets of a comparable scale to those used for general-purpose vision-language pretraining. Based on the literature reviewed in this work, research in this area can be divided into three complementary directions. The first focuses on geospatial foundation models (GeoFM) that extend vision-only pretraining to multi-sensor and geo-temporal data without incorporating a language component. The second adapts CLIP-style contrastive pretraining to curated remote-sensing image-text datasets in order to establish a direct connection between visual and linguistic representations. The third approach extends cross-modal reconstructive pretraining to remote-sensing data by combining natural-language captions with other modalities.

2.5.1 Language-free multi-sensor Foundation Models

SatMAE [13] and SpectralGPT [14], introduced in Section 2.2, are designed around individual sensors and acquisition settings. SkySense [25] extends the contrastive approach to a billion-parameter multimodal encoder that processes temporal sequences of spatially aligned WorldView RGB, Sentinel-1 SAR and Sentinel-2 optical imagery. Copernicus-FM [26] integrates a broader range of Copernicus Sentinel missions. These missions provide information on surface and atmospheric processes as well as non-imagery metadata. Both approaches demonstrate the feasibility of large-scale foundation encoders that integrate multiple types of Sentinel-family data. However, neither model incorporates natural language into its learned representation space. Language-based downstream applications therefore still require a separate alignment mechanism.

2.5.2 Contrastive FM adaptions to Remote Sensing

The second research direction holds on to the contrastive dual-encoder architecture and training objective of CLIP while replacing its natural images training data with assembled remote-sensing image-caption datasets. RemoteCLIP [27] combines existing captioning and detection datasets, RS-CLIP [28] adds pseudo-labelling and curriculum learning to improve zero-shot scene classification. GeoRSCLIP [29] scales to the five-million-pair RS5M dataset, and DOFA-CLIP [30] extends the backbone to different spectral configurations using dynamic hypernetworks and the six-modality GeoLangBind-2M dataset, which covers six heterogeneous modalities. The SAR captions are either focused on floods and ships or were extracted from longer RGB-based captions. Two more recent approaches decouple language alignment from image-caption pair datasets. Zavras et al. [31] extend CLIP to new sensor modalities by distilling knowledge from its pretrained visual and textual encoders, without requiring additional text data for the target modality. SATtxt [32] similarly transfers spectral knowledge to an RGB student through distillation, while replacing the CLIP text encoder with an instruction-tuned large language model as the target for visual-language alignment. Currently, it is limited to optical imagery with plans for future work to broaden the input to SAR and thermal imagery. Across these approaches, the resulting language alignment remains closely tied to the visual content represented in the respective training data. Since the underlying datasets are predominantly focused on optical imagery, the learned representations may provide limited coverage of concepts that are specific to other remote-sensing modalities.

For SAR imagery, several recent studies adapt the CLIP framework to SAR image-text data, combining SAR-specific caption datasets with different training procedures to address different SAR-specific challenges. SARCLIP by Wang et al. [33] targets the high-resolution input problem, while another model also named SARCLIP [34]

focuses on speckle robustness and multi-scale alignment. SARVLM [35] employs a two-stage domain-transfer strategy that uses optical remote sensing as an intermediate step between natural imagery and SAR. The dedicated caption datasets [30, 33, 34, 35, 36, 37] used in this line of research reflect a fundamental trade-off. Captions derived directly from SAR data preserve the modality-specific information but are often restricted to predefined templates. Captions generated from optical imagery can provide richer and more varied linguistic descriptions, but introduce information from a different modality.

2.5.3 Cross-modal reconstruction: TerraMind

TerraMind [38] uses an any-to-any generative architecture for Earth observation based on a shared symmetric Transformer encoder-decoder. On the PANGAEA benchmark [39], which includes nine segmentation datasets covering tasks such as flood mapping, crop segmentation, land-cover mapping, marine debris detection, and change detection. The authors report that TerraMind was the first geospatial foundation model to outperform both previous foundation-model approaches and task-specific U-Net baselines trained from scratch. The model processes tokens from multiple modalities within a shared network and supports Sentinel-1 SAR, Sentinel-2 imagery from L1C and L2A products, land-use and land-cover information, NDVI, digital elevation models, geographic coordinates, and natural-language captions. Image-based modalities are represented using finite scalar quantisation with modality-specific codebooks, while sequence-based modalities such as captions and coordinates are tokenised using WordPiece and generated autoregressively during decoding.

Pretraining is performed on TerraMesh, a globally distributed multimodal dataset containing around nine million spatiotemporally aligned samples of 264×264 pixels at a 10 m ground sampling distance [40]. TerraMesh builds on the Sentinel-1 and Sentinel-2 grids from SSL4EO-S12 v1.1 [41] and a subsampled version of MajorTOM-Core [42]. It is further enriched with ESRI LULC maps corrected using SEnSeI v2 cloud masks, Copernicus 30m DEM tiles, NDVI calculated from the Sentinel-2 red and NIR bands, discretised centre coordinates, and the LLaVA-Next captions discussed in Section 3.3 [40]. Since TerraMesh partially overlaps with the dataset used in this work, the SSL4EO-S12 training and validation splits are kept unchanged throughout the experiments to avoid data leakage when evaluating the frozen encoder.

The pretraining objective uses masked cross-modal reconstruction. Random target tokens are predicted from a separately sampled subset of input tokens and pixel-level information, resulting in a cross-modal patch-level classification task trained with cross-entropy loss. Because all modalities share the same encoder and the input and target subsets are sampled from a Dirichlet distribution, the representation of each modality is influenced by reconstruction targets from the other modalities [38, 43].

This also applies to Sentinel-1 and Sentinel-2 when natural-language captions are used as reconstruction targets, creating a direct training connection between the modalities. TerraMind is available in two encoder sizes, TerraMindv1-B and TerraMindv1-L, both trained on 500B tokens using 32 NVIDIA A100 GPUs [40]. TerraMindv1-B uses an embedding dimension of 768 with 12 Transformer encoder blocks, 12 attention heads, and an MLP ratio of 4 [44, terramind_register.py].

2.5.4 Remote Sensing VLMs

Alongside these encoder foundations, a range of remote-sensing VLMs adapts the projector-bridge architecture introduced in Section 2.4 to Earth-observation data and is able to create captions.

The optical EO-VLM literature is mainly based on LLaVA- and BLIP-style architectures, which primarily differ in their instruction-tuning datasets and task coverage. This group includes RSGPT [45], GeoChat [46], RS-LLaVA [47], SkyEyeGPT [48], and LHRS-Bot-Nova [49]. RingMoGPT [50] extends this architecture with a location-aware Q-Former and a dedicated change-detection module. SkySenseGPT [51] contributes a 1.8-million-sample dataset designed to support relation reasoning, while Marimo et al. [52] extend the projector-based approach beyond RGB imagery to full Sentinel-2 multispectral input.

SAR-specific captioning extensions generally retain established vision-language architectures and adapt them to SAR through dedicated fine-tuning datasets. He et al. [36] train two models, SAR-RS-CoCa for image captioning and SAR-GPT for visual question answering, on a dataset of 130 K SAR image-text pairs. Both models follow a two-stage adaptation strategy. They first continue pretraining their natural-image backbones on an optical remote-sensing dataset and then fine-tune the complete models on the SAR-text dataset.

Multi-sensor extensions expand instruction tuning across several remote-sensing modalities. EarthGPT [53] instruction-tunes a language model on MMRS-1M, which contains 1.01 M image-text pairs aggregated from 34 optical, SAR, and infrared datasets. EarthGPT-X [54] builds on the same vision encoders and adds visual prompting and pixel-level grounding. EarthDial [55] covers RGB, SAR, and infrared imagery. Its three-stage training procedure introduces multispectral and SAR data only during the final stage, while the natural-image vision encoder remains frozen. EarthMind [22] addresses the resulting modality imbalance by quantifying it with a Modality Attention Score. The authors show that models based on simple token concatenation consistently assign substantially less attention to SAR information across the language model layers. EarthMind therefore introduces a hierarchical cross-modal attention mechanism. For all these systems, the vision encoder is based on natural-image pretraining and does not provide a dedicated SAR-specific representation. The linguistic

supervision associated with SAR data is likewise derived either from label-based data aggregation or from vision-language models applied to optical imagery.

Across the reviewed multi-sensor models, SAR-CLIP variants, and SAR-specific captioning models, the visual encoder is always based on natural-image pretraining. The models either keep the encoder frozen and learn a projector on top of it or fine-tune the encoder on remote-sensing data.

2.6 Research Gap

The different groups of geospatial vision-language models reviewed in this chapter address different parts of the problem of connecting both Sentinel-1 and Sentinel-2 imagery with natural language. Each provides part of the required functionality, but the specific combination investigated in this thesis remains insufficiently explored.

A system that aligns Sentinel-1 representations with a large language model while avoiding an implicit dependence on optical imagery should satisfy three conditions simultaneously:

1. The visual representation should be shaped by Sentinel-1 data during pretraining.
2. The alignment target should consist of free-form VLM-generated descriptions, exploiting the semantic capacity of a modern LLM.
3. The linguistic supervision for the SAR data should be grounded in the SAR observation itself.

To the best of our knowledge, no existing system combines these three properties within a projector-bridge architecture.

3 Data

3.1 Sentinel-1 and Sentinel-2

Remote sensing refers to the acquisition of information about an object or area by a device that is not in contact with it [56]. In Earth observation this is realised through the measurement of electromagnetic radiation reflected or emitted from the surface. Sensors are divided according to their energy source. Passive systems record naturally available energy, mainly reflected solar radiation, and therefore depend on external illumination. Active systems supply their own energy, illuminate the scene themselves, and are consequently able to operate at night and through cloud cover [57, pp. 18, 54–56]. Both families are represented in the Copernicus space segment by Sentinel-1 and Sentinel-2, the two data sources used in this work.

3.1.1 The Copernicus open data policy

The decisive precedent for open Earth observation archives was set in 2008, when the U.S. Geological Survey released the entire Landsat archive free of charge, which led to a rapid increase in scientific use [58]. Its European counterpart is the Copernicus programme, implemented by the European Union together with its Member States and the European Space Agency, to which the technical coordination of the space component is delegated [59]. Under its free, full and open data policy, any user may reproduce, distribute and modify Copernicus data worldwide [60, Art. 49]. Since 2023 the Sentinel data have been distributed through the Copernicus Data Space Ecosystem [61]. This policy, combined with the systematic global acquisition strategy of the Sentinel missions, is what makes globally distributed, densely sampled and co-located SAR–optical dataset available at all, and therefore forms the practical precondition for the dataset.

3.1.2 Sentinel-2: passive multispectral imaging

Sentinel-2 carries the Multi Spectral Instrument, a passive push-broom sensor that samples thirteen spectral bands from the visible and near infrared to the short-wave infrared at spatial resolutions between 10 and 60 m over a swath of 290 km [62]. Sentinel-2 measures reflectance, which is mainly related to the chemical composition and structure of the observed material [63, p. 44]. The wavelength-dependent reflectance of a surface forms its *spectral signature*, which is often indicative of surface type or condition and thereby provides the basis for feature identification and

classification. Figure 3.1 shows this behaviour for four representative surface types together with the Sentinel-2 band passes. A single broad band would average the variation out, whereas multiple narrower bands retain the characteristic shape of the reflectance curve. Separability is nevertheless not guaranteed, since materials with very different physical properties may exhibit similar spectral responses [56, pp. 21, 46]. This work uses the Level-2A product, which provides atmospherically corrected surface reflectance [64]. This makes the observations more comparable across different acquisition conditions, while avoiding the need for the model to learn variations caused by atmospheric conditions across different dates and locations.

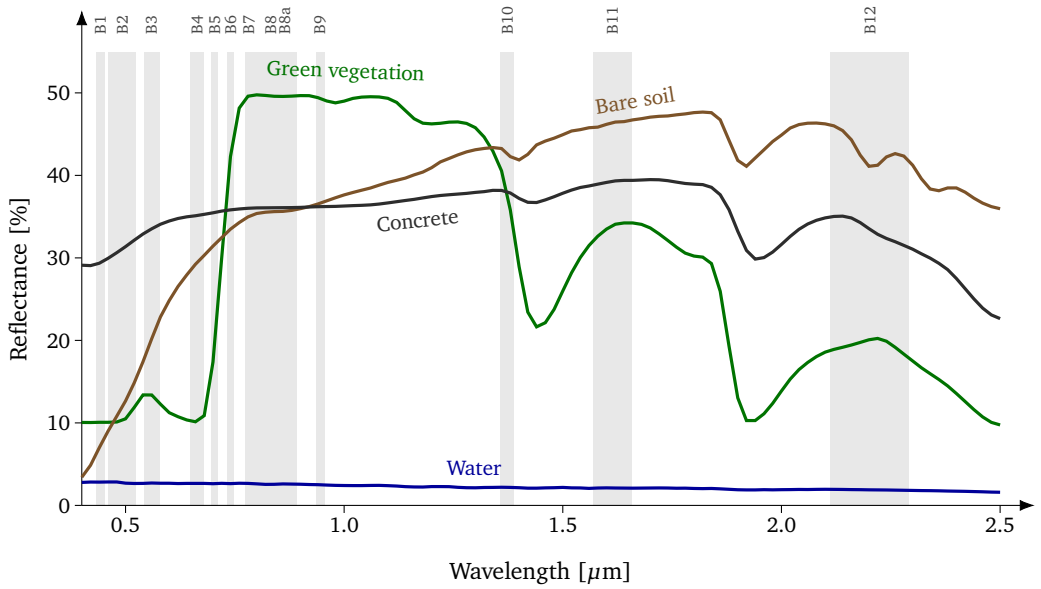


Figure 3.1: Spectral reflectance signatures of four surface types and the thirteen Sentinel-2A band passes shown in grey. Shaded intervals indicate the wavelength range of each band based on its full width at half maximum [64], narrow bands are shown with a minimum width for readability. Reflectance curves are based on measurements from the ECOSTRESS Spectral Library [65, 66] for red maple foliage, brown sandy loam, concrete, and tap water.

3.1.3 Sentinel-1: active microwave imaging

Each Sentinel-1 satellite carries a single C-band synthetic aperture radar with a centre frequency of 5.405 GHz, corresponding to a wavelength of approximately 5.55 cm [67]. The instrument emits a controlled microwave signal and measures the backscattered return, from which the surface scattering properties are retrieved [57, p. 204]. Since a real aperture would provide insufficient azimuth resolution from orbit, SAR uses the motion of the platform to create a synthetic aperture. This allows an azimuth resolution of roughly half the physical antenna length, largely independent of the range [68]. The measured quantity is the backscattering coefficient σ^0, the average radar cross section per unit ground area, which spans several orders of magnitude and is therefore expressed on a decibel scale. It depends on system properties such as wavelength, polarisation, and viewing geometry, as well as on properties of the observed surface,

including roughness, moisture, and vegetation [57, pp. 222–223]. The corresponding idealised mechanisms are sketched in Figure 3.2. Surfaces that are smooth relative to the wavelength reflect specularly and appear dark, which is why calm water bodies are dark in radar imagery, whereas two perpendicular smooth surfaces form a corner reflector whose double-bounce return makes urban areas bright. A vegetation canopy acts as a volume of randomly distributed scatterers and depolarises the signal, so that the cross-polarised channel of the dual-polarised VV+VH configuration separates volume scattering from surface scattering [56, pp. 424–425]. The interpretability of the signal is, however, limited due to different properties. Speckle is a pseudo-random variation in brightness caused by interference within a resolution cell. It affects all pixels and has a multiplicative effect, so multi-look processing can reduce it, but cannot completely remove it [68, p. 12]. Image intensity also depends strongly on the incidence angle and viewing direction. Therefore, the idea of a spectral signature from optical imagery cannot be directly applied to radar data [57, p. 223].

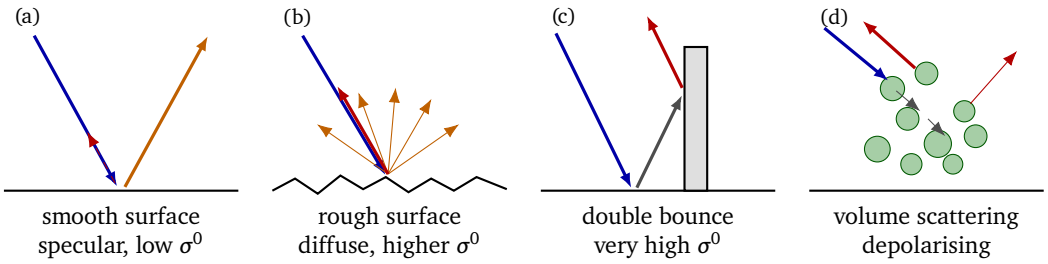


Figure 3.2: Idealised scattering mechanisms determining the backscatter coefficient σ^0. Blue arrows denote the incident signal, red arrows the component returned towards the sensor. (a) A surface that is smooth relative to the wavelength reflects specularly and appears dark; (b) a rough surface scatters diffusely and returns a larger share of the energy; (c) adjacent smooth surfaces produce a double reflection with a very high return; (d) a vegetation canopy acts as a volume of discrete scatterers, which depolarises the signal and thus raises the cross-polarised return. Compiled from [56, Fig. 6.26], [57, Fig. 7.18] and [63, Fig. 6.1].

3.1.4 Complementarity and cloud dependency

All radiation detected by a spaceborne sensor passes through the atmosphere, where it is altered in intensity and spectral composition by scattering and by absorption at the characteristic wavelengths of water vapour, carbon dioxide and ozone, which confines optical bands to the transmissive ranges known as atmospheric windows. Clouds remove the optical signal entirely, whereas the 5.55 cm wavelength of the C-SAR is unaffected by cloud cover and independent of solar illumination [57, pp. 21, 54–56], so that Sentinel-1 provides observations where Sentinel-2 cannot. This is the *robustness* objective of image fusion, in which unavailable or unreliable observations in one modality are replaced by information from another, the standard examples being cloud-covered regions in optical imagery and radar shadows in SAR data. In contrast, the second objective applies even when both sources are available. Because the two modalities respond to different physical properties of the surface, their combination

can reveal patterns that are not apparent in either dataset alone and thereby reduce ambiguity in interpretation [69, pp. 827–828]. Data fusion integrates measurements from multiple sources to obtain information that cannot be derived from any individual sensor alone. Depending on the stage at which the information is combined, observation-level, feature-level and decision-level fusion are distinguished. But to achieve that, the sources must be brought into a common reference frame, which is demanding for SAR and optical data because the two differ substantially in geometric and radiometric characteristics [70].

3.2 SSL4EO-S12 v1.1

The project uses satellite imagery from SSL4EO-S12. This dataset was created specifically for self-supervised learning during pre-training [71]. The original version of the SSL4EO-S12 dataset is available on mediaTUM [72].

The satellite imagery used in this work was obtained from the Sentinel-1 and Sentinel-2 missions through the Google Earth Engine Data Catalog. The Sentinel-2 data are based on Level-2A products, which provide atmospherically corrected surface reflectance scaled to digital numbers [64, 73]. Since the bands have different native spatial resolutions, all of them were resampled to 10 m using nearest-neighbour interpolation [41]. The Sentinel-1 data consist of GRD products acquired in interferometric wide swath (IW) mode with VV and VH polarisation [71]. These products contain calibrated and terrain-corrected backscatter values converted to dB [74, 75]. The data are provided on a 10 m pixel grid, although the effective spatial resolution is approximately 20 m [67]. An overview of the main characteristics of both modalities is given in Table 3.1.

The SAR data still contain several effects that are relevant for their interpretation. Multi-looking reduces speckle, but no additional filtering is applied [64]. Radiometric terrain flattening is also not applied [75]. As a result, slopes facing towards or away from the sensor and radar shadows remain visible in the data [56, p. 418]. The same effects are present in the false-colour composites used to generate the SAR captions in Section 3.3.

Regarding the temporal dimension, acquisitions were selected around the astronomical season onsets in 2021, with imagery from 2020 serving as an additional temporal buffer. The exact acquisition dates were chosen based on image quality criteria, in particular ensuring a cloud coverage of less than 10% for the Sentinel-2 observations. Beyond seasonal diversity, the dataset also incorporates geographic variability by sampling the 10,000 largest cities worldwide and their surrounding regions, resulting in near-global coverage, as visible in Figure 3.3. [71]

The extracted image patches are 264×264 pixels, and strongly overlapping patches

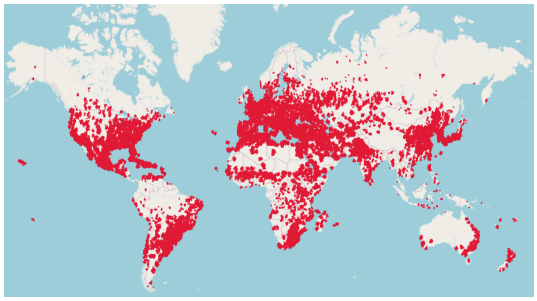


Figure 3.3: Spatial distribution of the SSL4EO-S12 dataset samples globally. Taken from [71]

were removed during preprocessing. In total, approximately 250,000 geographic locations were included. Combining the four acquisition periods with the three available image types results in roughly three million patches [71].

In the revised version v1.1, geospatial alignment inaccuracies between modalities were resolved, and the data structure was revised. It was taken care to remain as close as possible to the original data points in terms of time and location. Locations with more than 1% missing values for a given time point or modality were excluded. To make the dataset suitable for extended multimodal pretraining, modalities for elevation, land cover, and vegetation were also added, but these are not relevant to this study. [41]

Modality	**Sentinel-1 GRD**	**Sentinel-2 L2A**
Type	SAR	Optical
Number of Bands	2	12
Range	-50 - +1	0 - 10k
Unit	dB	DN

Table 3.1: Summary of modalities included in SSL4EO-S12 v1.1 and used in this work, based on [41]

The dataset is split into training and validation sets at a 99:1 ratio, with no overlap between the two. The modalities are stored in separate folders [41]. The samples were compiled into ZARR files (ZARR version 2), with each ZARR file containing 64 locations at 4 time points. The data were divided into blocks by time to allow for easy use with just a single time point. The files can be downloaded from the Forschungszentrum Jülich Data Publication Server, a web server that makes large datasets available to the general public with a focus on scientific publications. [76]

The dataset is well suited for this project because it provides co-registered Sentinel-1 and Sentinel-2 acquisitions of the same scenes, as visible in Figure 3.4. Even though the two modalities are treated separately in this work, the co-registration is still useful because both modalities describe the same scene, which allows the S1 and S2 results to be compared directly, at least on a manual level. In addition, image descriptions are already available for these image pairs, which are discussed in the following section. Google Earth Engine (GEE) updates the collection daily, with an ingestion

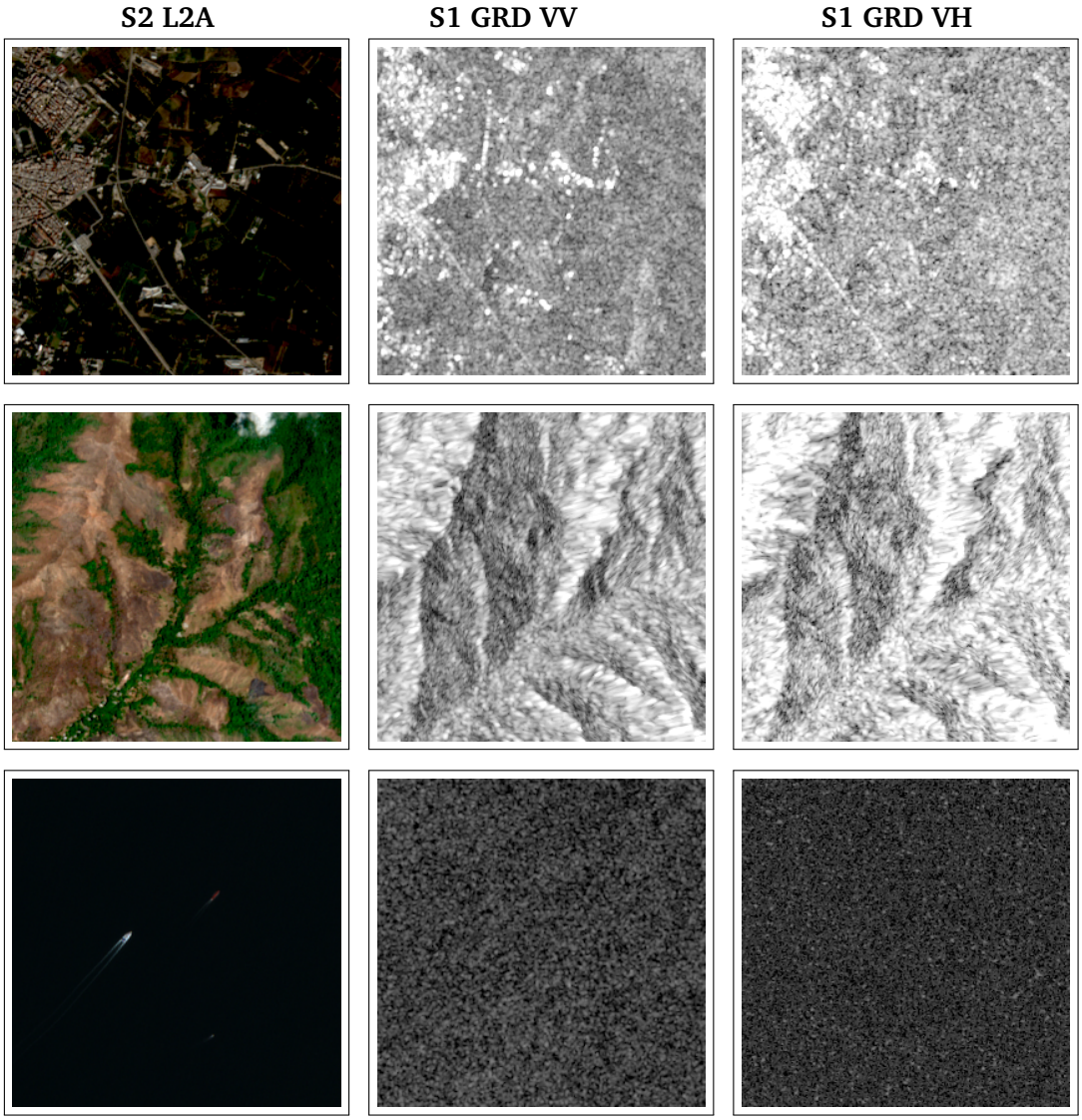


Figure 3.4: Examples of the SSL4EO-S12 dataset. Sentinel-2 L2A is shown as true-color RGB. Sentinel-1 GRD is shown as separate VV- and VH-polarizations in dB scale. [76].

delay of approximately two days [77]. Consequently, newly acquired satellite imagery becomes available with the same preprocessing pipeline within a few days, facilitating the straightforward transfer of the project to current and future applications.

It should be noted that this dataset was already used during the pretraining of TerraMind, which represents a potential source of contamination. However, as no alternative dataset provided the same level of quality while fully satisfying the project requirements, this dataset was nevertheless selected for this work. This choice can be justified by freezing the pretrained encoder during training. As a result, the trainable components of the proposed model have not previously been exposed to these data, and no data leakage is expected.

The dataset is published on Hugging Face under the permissive CC BY 4.0 license, which permits its use in this work [41].

Since no test split is provided in the dataset, the validation split was further divided into val and test. The resulting dataset partition comprises 60 ZIP files assigned to the training set, 10 ZIP files to the validation set, and 20 ZIP files to the test set.

3.3 Automated Caption Generation for SSL4EO-S12

3.3.1 Llama3-SSL4EO-S12

The S2L2A captions used in this work originate from the Llama3-SSL4EO-S12 dataset [52]. The dataset contains around one million natural language descriptions associated with images from SSL4EO-S12 v1.1. Its main purpose is to provide semantic information for contrastive vision-language pretraining. This is useful for satellite imagery because such data usually do not come with detailed textual descriptions. Instead, many Earth observation datasets mainly provide class labels, metadata, or similar structured information. [52]

The captions were generated with the multimodal large language model Llama3-LLaVA-Next-8B. The generation process combines information extracted from the satellite image with additional geographic data. The image input consists of the RGB bands from Sentinel-2 Level-2A products. These bands were converted to 8-bit values and resized to 224×224 pixels before being provided to the model. Additional geographic context was obtained from the Overture Maps base layer. This information can provide details about land cover, infrastructure, and other mapped objects in the area. Objects smaller than 2500 square metres were not included in this context. The model was then guided by a structured prompt. As an intermediate step, three question–answer pairs describing relevant aspects of the scene were generated. These were subsequently used as additional input for the caption generation. The prompting procedure also contains further restrictions intended to reduce hallucinated information. [52]

The quality of the resulting captions was examined using automatic measures as well as a manual evaluation. The captions of the validation split showed a higher average n-gram diversity (0.75) than the manually labelled datasets RSITMD, RSICD and UCM Captions (0.48–0.49). The pairwise similarity between captions was also lower, indicating greater lexical variety. For the manual assessment, fourteen researchers with an Earth observation background reviewed approximately 1,300 captions selected at random from the validation split. They evaluated whether the captions covered the relevant image content and marked cases in which information was hallucinated. More than 85% of the evaluated captions were judged to describe most of the relevant content adequately. Hallucinated information was found in about one third of the samples, although these cases usually concerned only a single feature per image.[52]

The dataset authors identify hallucinations as a limitation and report several types of errors. These include describing land features that are not present in ocean areas, incorrectly identifying anthropogenic or aquatic structures, and giving incorrect numbers or sizes of objects. Such errors and other biases in the captions can also affect models trained on this data. The authors further note that the vocabulary is concentrated on frequently occurring topics and that more linguistic variety would be desirable. For applications where incorrect descriptions cannot be tolerated, they therefore suggest adding human verification to the caption generation process. Despite these limitations, the dataset contains scene-specific information such as snow cover, cloud conditions, and colour, which is less strongly represented in datasets such as ChatEarthNet and SkyScript. [52]

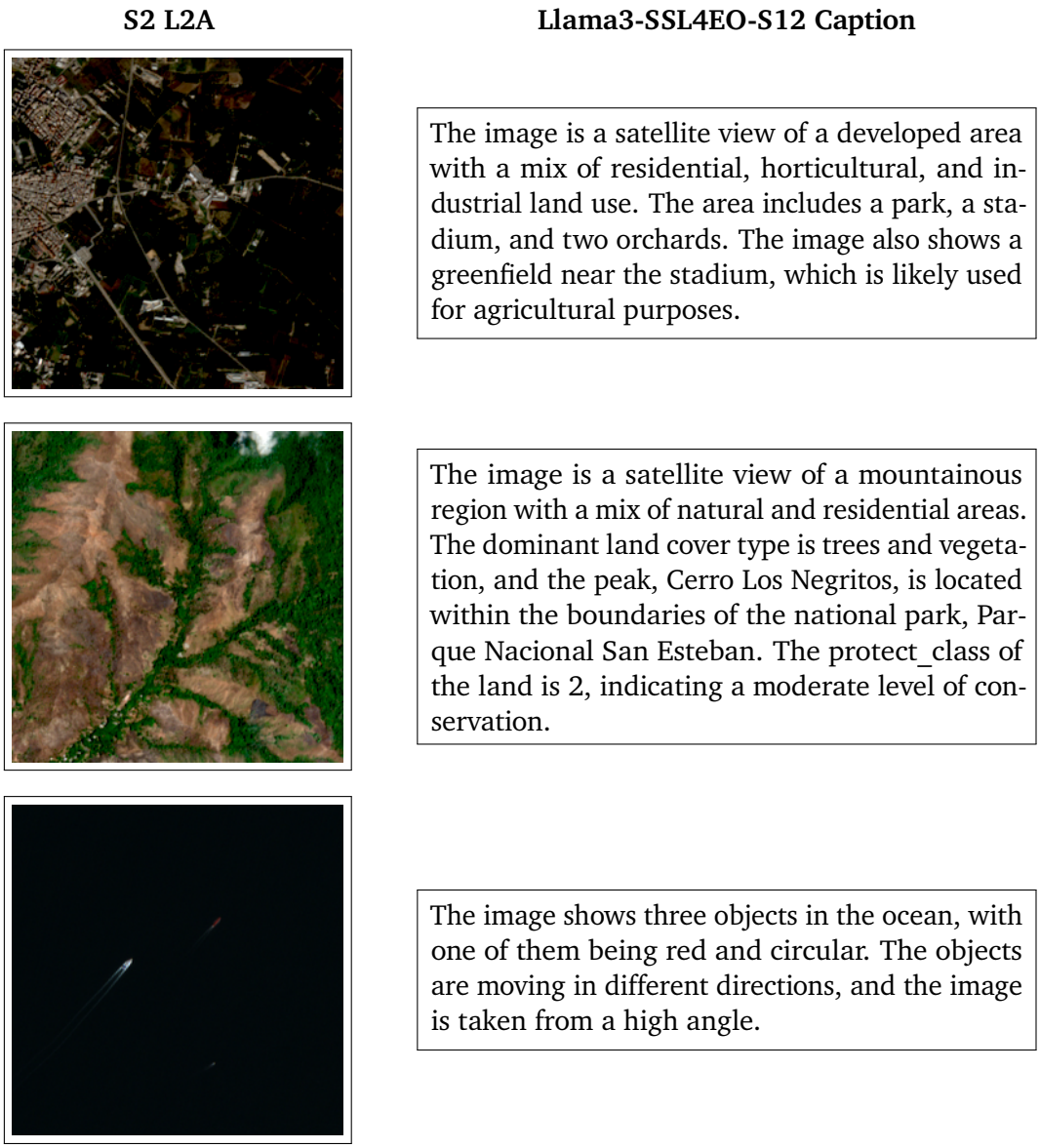


Figure 3.5: Examples of the SSL4EO-S12 dataset with the corresponding Llama3-SSL4EO-S12 captions. Sentinel-2 L2A is shown as true-color RGB and the upper two images have been brightened for better visibility, just for this presentation. Taken or adapted from [41] and [52].

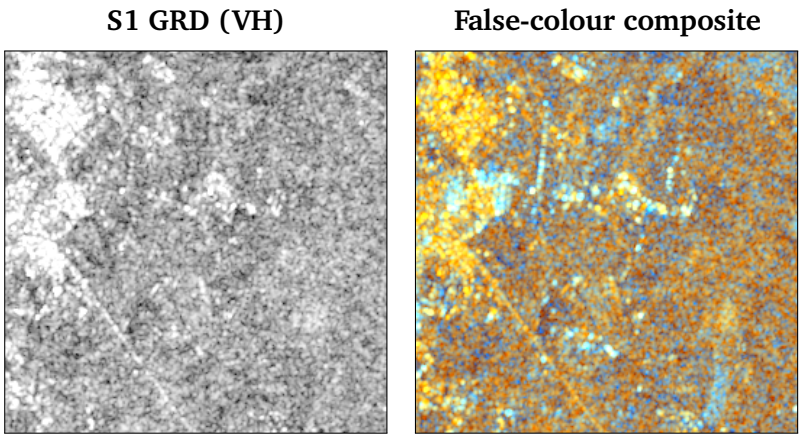
For this work, it is relevant to note, that the captions were generated exclusively from the RGB representation of the Sentinel-2 L2A data [52]. The multispectral channels and the co-registered Sentinel-1 GRD data contributed nothing to the description, so the linguistic supervision describes optically observable scene content rather than radar-observable surface properties. The authors themselves point to the alignment with the Sentinel-1 GRD data as unexploited potential of the dataset, and the reported completeness and hallucination rates define a limit on the supervision quality available to any model trained on it [52]. Figure 3.5 shows three examples from this dataset together with their corresponding captions. The scenes were selected to cover a broad range of land cover types, including an urban area, a natural area, and an ocean scene. The selected scenes have the global IDs 4838, 10399, and 111802.

3.3.2 SAR captions

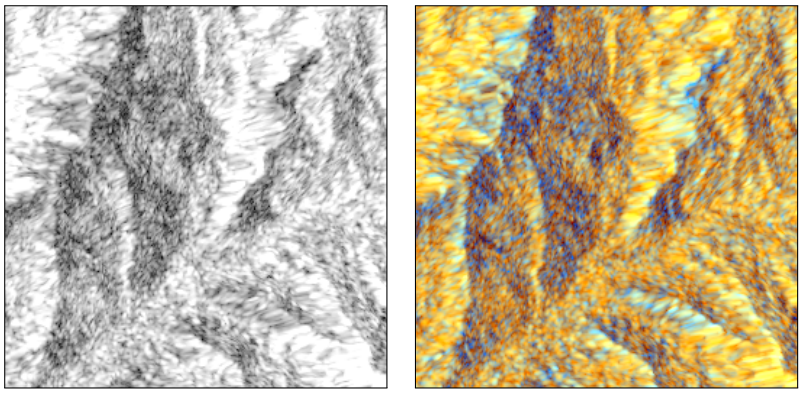
To overcome the missed potential of SAR-specific information in the captions and since none of the six SAR-text datasets mentioned in Section 2.5 provides a cleanly extractable Sentinel-1-based subset with free-form VLM-generated descriptions, we have created our own caption dataset based on the S1GRD data from the SSL4EO-S12 v1.1 imagery dataset.

Pseudo-RGB composites were generated from the dual-polarised Sentinel-1 GRD tiles. The backscatter values are kept in decibels because the corresponding intensity values are approximately log-normally distributed, and a direct linear conversion would assign only a small portion of the output range to low-backscatter classes. Each channel was subsequently converted to 8 bit using a fixed affine transformation, $v = 255 \cdot (x - lo)/(hi - lo)$, with the resulting values clipped to `[0, 255]`. The limits `lo` and `hi` were estimated once based on approximately 5,000 images from 20 training batches and then kept unchanged for all tiles, including the validation data. Per-tile normalisation was therefore not applied, since it would make grey values incomparable between scenes. VH was assigned to the red channel, VV to the green channel, and the difference $VV - VH$ to the blue channel. The last one represents the co- to cross-polarised ratio in the dB domain and acts as a depolarisation indicator, providing additional information for distinguishing surface and volume scattering. Pixels with invalid values in either polarisation were assigned zero in all three output channels. We then prompted Gemini-2.5 Flash-Lite [78] to generate a textual description of each scene based on this SAR-derived false-RGB representation. The prompt is listed in Chapter A.

The captions shown as examples in Figure 3.6 are not required to match those generated from Sentinel-2 or its RGB bands. Differences are expected because the two sensors capture the scene using different imaging principles. In addition, the Sentinel-1 and Sentinel-2 acquisitions may have been taken at different times, meaning that

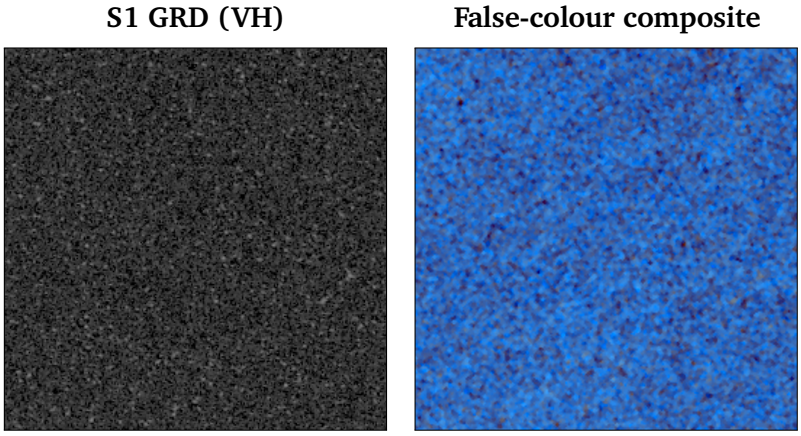


This landscape is characterized by a mix of high-volume 3D canopy biomass and low-biomass ground cover, presenting a textured, heterogeneous surface. In the upper-left quadrant, a concentrated cluster of intense, bright white double-bounce structural reflections indicates the presence of an isolated rural settlement or farmstead. The remainder of the frame is dominated by muted, desaturated tones representing transitioning land use, punctuated by faint, linear geometric boundaries. These subtle structural signatures indicate a landscape under active human management, contrasting against the diffuse, grainy texture of the surrounding natural vegetation.



A rugged, undulating landscape dominated by expansive regions of high-volume 3D canopy biomass, characterized by vibrant gold and orange textures. These forested zones are dissected by prominent, sinuous dark blue topographical depressions, indicative of natural drainage patterns or deep valley systems. Throughout the frame, particularly along the lower and lateral margins, subtle geometric variations in surface texture suggest managed terrain, though the area is primarily defined by its dense, organic natural cover rather than engineered urban infrastructure. The absence of intense, clustered double-bounce structural reflections confirms this as a remote or minimally developed wilderness environment.

Figure 3.6: Examples from the SSL4EO-S12 test split with the corresponding Gemini-generated captions. Left: VH backscatter in dB. Right: false-colour composite (R: VH, G: VV, B: VV-VH) from which the captions were generated. Adapted from [41].



This landscape is characterized by a uniform expanse of low-biomass ground cover, represented by a saturated blue-hued surface texture indicative of open terrain or pasture. The image lacks distinct geometric man-made boundaries or high-intensity double-bounce structural reflections, suggesting an absence of built infrastructure or settlements within this specific frame. The uniform surface quality points to a consistent, featureless natural environment, devoid of dense 3D canopy biomass or complex topographic variation.

Figure 3.6: Examples of the Gemini-generated S1 captions (continued).

the observed scene can also differ slightly between the two images. A comprehensive evaluation of these image-caption pairs by several SAR experts was not possible within the scope of this thesis, but it reached an average n-gram diversity of 0.82, which is higher than in the Llama3-SSL4EO-S12 dataset. Additionally, a SAR expert has inspected parts of the dataset and rated them as valid.

3.4 Data Handling

The dataset is organised into train, validation, and test splits using a directory-based structure. Within each split, the data is divided into the categories S1GRD, S2L2A, and captions. Satellite images are stored as Zarr ZIP files, while the corresponding captions are provided as CSV files, as illustrated in Figure 3.7. Images and captions are matched at the shard level based on their directory location and consistent file naming.

Within each shard, an additional local indexing scheme is required to access individual images. Zarr stores the data as multidimensional arrays. For this dataset, the arrays have five dimensions, (B, T, C, H, W), where B denotes the batch dimension, T the temporal dimension, C the number of channels, and $H \times W$ the spatial dimensions [76]:

$$x_{S1} \in \mathbb{R}^{64\times4\times2\times264\times264}, \qquad x_{S2} \in \mathbb{R}^{64\times4\times12\times264\times264}$$



Figure 3.7: Dataset directory structure (shown exemplarily; test/ follows the same structure as val/).

Samples are accessed using position-based indexing, meaning that the requested element is determined by its offset within the underlying array. Therefore, valid local indices range from 0 to 63. However, the `sample_id` values stored in the CSV files are assigned globally across the SSL4EO dataset and can be much larger than 63. They are also not ordered by shard. To map the CSV entries to the corresponding positions in the Zarr arrays, the global IDs are replaced by local indices based on their order of first occurrence within each shard. Empirical checks confirmed that the sample order in the Zarr arrays corresponds to the order of entries in the associated CSV files.

The preprocessing pipeline is implemented as a PyTorch map-style `Dataset`. After initialisation, each sample is represented as a dictionary containing `sample`, `time`, `caption`, `caption_s1`, `tags`, `classes`, `s1_zarr`, and `s2_zarr`. When `dataset[idx]` is called, the image is optionally center-cropped according to the selected configuration. This preprocessing determines the number of visual tokens passed to the encoder, which increases quadratically with the image resolution.

Each spectral band is then normalised separately using the fixed per-band statistics provided with the TerraMind checkpoint [44]. These statistics are required to match the input distribution expected by the frozen encoder. The band order follows SSL4EO-S12 v1.1, as described in Section 3.2. Sentinel-2 L2A uses the twelve bands B01–B08, B8A, B9, B11, and B12 in raw reflectance units, while Sentinel-1 GRD uses the VV and VH polarisations in dB.

Finally, the preprocessed images and their associated captions, tags, and classes are passed to a custom `collate_fn`. The function combines the individual samples into batched tensors and returns them as a dictionary. The output contains the same keys as the individual samples, with tensors and lists stacked along the first dimension according to the configured `batch_size`.

4 Methodology

4.1 Overview

This chapter describes the pipeline used to generate natural language descriptions from Sentinel imagery and the experimental setup used to evaluate it. The main design choice is to keep both the pretrained EO encoder and the pretrained language model frozen and to train only a small module between them. This setup allows the experiments to focus on the alignment between the visual representation and the language model instead of an end-to-end optimisation of the complete system. The pipeline is visualized in Figure 4.1 and consists of three main components. A frozen visual encoder produces patch embeddings from a Sentinel scene. A trainable bridge maps these embeddings into the input embedding space of the language model. The frozen language model then combines the projected visual tokens with a short instruction prompt and generates the caption autoregressively. The visual encoder remains fixed throughout all experiments, while the bridge and language model are varied according to the experimental setup.

The reference caption is first tokenised and passed through the frozen input embedding of the language model. These caption embeddings are then combined with the visual tokens and the prompt. The next-token cross-entropy loss is evaluated only for the caption tokens. During backpropagation, the gradients pass through the frozen language model to the bridge.

The bridge is implemented either as a linear projection or as a two-layer MLP, allowing the effect of a small amount of non-linearity to be evaluated. For the language model, we compare a plain instruction-tuned language model with the language component of a vision-language model that has additionally undergone visual instruction tuning. This comparison allows us to examine whether prior visual instruction tuning improves the model's ability to use the projected Sentinel features.

Sentinel-1 and Sentinel-2 are processed in separate runs throughout the experiments. Their token sequences are not fused, so each result can be attributed to a single sensor modality. This separation also provides a controlled setting for comparing the language alignment achieved from SAR and optical representations.

The remainder of the chapter follows the individual components of the pipeline and builds on the data description given in the previous chapter. Section 4.2 introduces the visual encoder. Section 4.3 introduces the two language model settings. Section 4.4 describes the two bridge variants, and Section 4.5 the training objective and

optimisation setup.

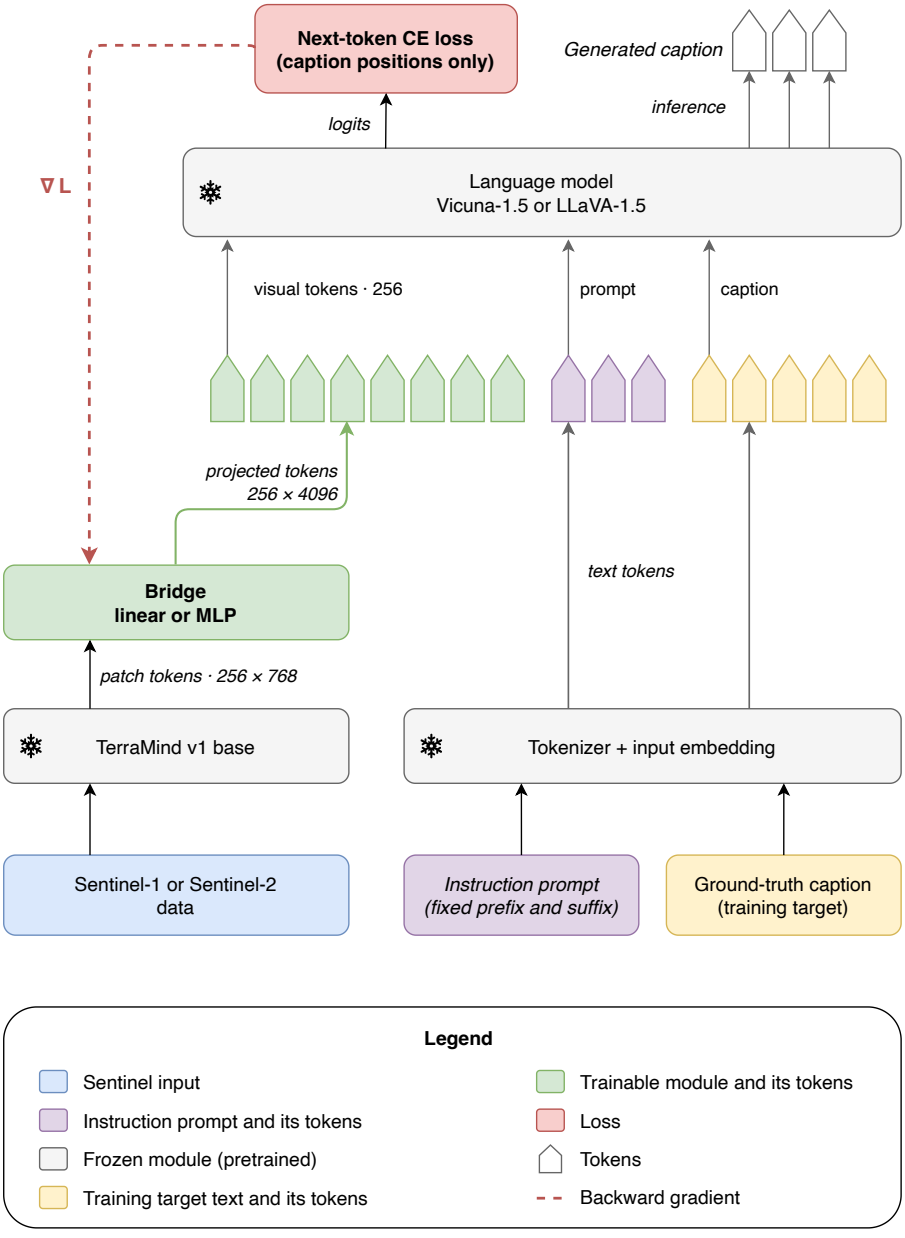


Figure 4.1: Alignment pipeline and training loop. A Sentinel-1 or Sentinel-2 scene is encoded by the frozen TerraMind v1 base encoder into 256 patch tokens (768-d). The trainable bridge (linear or MLP) projects them into the 4096-d input space of the frozen language model, where they are combined with the embedded instruction prompt and, during training, the embedded ground-truth caption. The next-token cross-entropy loss is computed on the caption positions only, and its gradient (dashed) updates the bridge alone. At inference, the caption is generated autoregressively from the visual tokens and the prompt.

4.2 TerraMind Encoder

This work uses the frozen encoder of TerraMindv1-B, introduced in Section 2.5, as its vision backbone, while the TerraMind decoder and the diffusion decoders of the tokenisers are not used, as the downstream task in Section 4.4 operates directly on the encoder representations and does not require the reconstruction of any modality. The role of TerraMind in this pipeline is therefore not its generative capability, but the intermediate representation.

The encoder is used in a setting that differs considerably from the one for which it was originally trained. During pretraining, the learned representations were used to predict information from other modalities, whereas in this work they have to be interpreted by a language model that has not been trained on Earth observation data. The pretraining objective does not directly ensure that the representations are suitable for this purpose, which is one of the main questions investigated by the experiments.

As described in Section 4.1, the S-1 and S-2 tracks are trained independently. Tokens from the two modalities are therefore never processed together in a single encoder pass, and this separation is already present in the first layer of the encoder.

4.2.1 Checkpoint loading and freezing

The encoder is loaded from the Hugging Face repository `ibm-esa-geospatial/-TerraMind-1.0-base` using the Terratorch backbone factory [44]. After initialisation, all encoder parameters are set to `requires_grad=False`, and the module remains in `eval()` mode throughout training. In addition, the encoder is evaluated within a `torch.no_grad()` block. This avoids storing activation graphs for the encoder blocks and therefore reduces memory usage. Together, these three measures ensure that the encoder produces a deterministic output for a given input crop, both across different epochs and across repeated runs. It also ensures that all bridge and decoder variants are evaluated using the same encoder representations.

4.2.2 Input resolution and token layout

All tiles are passed to the encoder at an input size of 256×256 pixels. This is the largest square that can be taken from the original 264×264 SSL4EO-S12 crops without upsampling. Therefore, the measurements are passed to the encoder at their original ground sampling distance, and no interpolation is applied to the input data. With TerraMind's fixed patch stride of 16 pixels, the crop can be divided evenly into a 16×16 grid, resulting in 256 token embeddings with a dimension of 768 for each image-like modality.

Since the Sentinel modalities are harmonised to a 10 m ground sampling distance [40], each token represents a physical area of 160 m × 160 m, while the complete tile covers 2.56 km × 2.56 km. This determines the level of detail at which information can be passed to the language model. A token represents a fixed square area of the ground that is approximately comparable in size to a large agricultural field or a city block. Features that are smaller than this area are therefore represented together with the other content within the same patch. However, the captions used to train the pipeline describe the scenes at a more general level, which makes this spatial resolution suitable for the task.

The positional embeddings provided with the checkpoint are originally defined for a 14 × 14 grid and are resampled to 16 × 16 during the forward pass. No CLS or register tokens are added (`num_register_tokens=0`), so all 256 output vectors correspond directly to physical patches in the input image and remain in the raster order of the patch grid. The bridge can therefore use the encoder output directly as a spatial feature representation without removing, pooling, or reordering any of the positions. This is an important requirement for the position-wise projection described in Section 4.4.

4.2.3 Modality handling

Sentinel-2 L2A contains 12 channels, while Sentinel-1 GRD contains 2 channels, so a single patch embedding cannot be used for both modalities. TerraMind therefore uses a separate input stem for each modality, and the backbone is constructed according to the modality selected for the respective run. The separation between the optical and SAR tracks is consequently not limited to the data loader, but is also present in the first layer of the encoder, where the two sensors are processed with separate weights.

The backbone returns its output as a dictionary indexed by modality, and the implementation concatenates the entries along the token axis according to the order specified in the modality list. In the experiments presented in this thesis, the dictionary contains only one entry, so the concatenation is straightforward and results in the 256-token sequence of the respective sensor. The same implementation can also process both modalities at the same time, in which case the resulting sequence would contain twice as many tokens. However, no such run is included in the experiments of this thesis.

4.2.4 Forward pass

The encoder maps each tile with a size of $C \times 256 \times 256$ pixels to a sequence of 256 token embeddings with a width of 768. Here, C is 12 for Sentinel-2 L2A and 2 for Sentinel-1 GRD.

The encoder provides the output of each of its blocks, so it is necessary to decide which representation should be passed to the bridge. In this work, the output of the last block is used, following the recommendation provided for the released checkpoint [44]. The LLaVA models use the layer before the last one of their CLIP vision tower instead. This choice is based on the assumption that the final CLIP layer focuses more on global image properties, while the preceding layer preserves more local information [23]. This reasoning is related to an encoder that was trained to align a complete image with a complete caption. TerraMind, however, is trained with masked cross-modal reconstruction at the level of individual patches [38], so the same argument does not apply to this setting. No comparison between different encoder depths was performed in this work, meaning that the choice is based on this reasoning and the recommendation for the released checkpoint. The resulting representations are passed directly to the bridge.

4.3 LLM Decoder

The language decoder is based on the frozen 7B-parameter LLaVA-1.5 checkpoint [21], which uses a Vicuna v1.5 causal language model as its underlying language model. Vicuna v1.5 itself was used as a variation of the pipeline in the experiments in Section 5.2.1. In this work, LLaVA-1.5 is used only as a language head. All language-model parameters remain frozen throughout training.

4.3.1 Checkpoint loading and freezing

The decoder is loaded from the Hugging Face repository `llava-hf/llava-1.5-7b-hf` using `LlavaForConditionalGeneration` of the `transformers` library [79]. After loading, the built-in `vision_tower` and `multi_modal_projector` modules are set to `None`, since visual information is provided exclusively through the bridge. All remaining parameters are then set to `requires_grad=False` using PyTorch's autograd [80]. The input-token embedding layer of the causal language model is accessed separately as `self.embed`. Its hidden size is 4096, which defines the output dimension that the bridge must produce. As the language model remains frozen, its dropout and layer-normalisation behaviour also remain unchanged during training.

4.3.2 Prompt template and prompt-embedding cache

Following the LLaVA-1.5 conversational format [21], each sample is provided to the language model as a single-turn dialogue with a fixed prompt surrounding the visual tokens:

```
"USER: <image tokens>  Describe the surface features
shown in this satellite observation. ASSISTANT:"
```

`<image tokens>` indicates the position where the output of the bridge is inserted. The proposed pipeline directly inserts the visual embeddings into the input embedding sequence using `inputs_embeds` from the Hugging Face API.

Since the prompt is identical for all samples, the prefix `USER:` and the remaining suffix are tokenised and embedded only once during initialisation. The resulting embeddings are stored as non-trainable buffers named `prefix_emb` and `suffix_emb`. This avoids repeatedly computing the embeddings of the same prompt tokens during training. The default instruction in the prompt can be easily changed without the need for further adaptions to the code. Variations for different purposes are listed in Chapter A.

4.3.3 Training objective and sequence assembly

For a batch of B tiles, the visual tokens produced by the bridge have the shape $B \times 256 \times 4096$ and are inserted into the prompt embedding sequence together with the corresponding ground-truth caption. The resulting input to the language model is the concatenation

$$e = [\, e_{\text{pre}},\ v,\ e_{\text{suf}},\ e_{\text{cap}} \,] \in \mathbb{R}^{B \times L \times 4096}, \qquad L = L_{\text{pre}} + 256 + L_{\text{suf}} + L_{\text{cap}},$$

where e_{cap} is obtained by tokenising each caption with the LLaVA tokenizer, appending an EOS token, and right-padding the sequences. The resulting token IDs are passed through `self.embed` to obtain the corresponding embeddings. Attention masks are set to one for all non-padded positions.

The label sequence is constructed so that only the caption tokens contribute to the training loss. The positions corresponding to the prefix, visual tokens, suffix, and paddings are filled with the PyTorch ignore index -100. The language model then computes the standard next-token cross-entropy loss over the shifted labels [15]. The Hugging Face implementation averages the loss over the remaining caption tokens only. The forward pass is performed with a `torch.autocast`. Since the language-model parameters θ are frozen, the gradient of the loss propagates only through the bridge.

4.3.4 Inference

During inference, the caption is removed from the input sequence, so the language model receives only the prompt prefix, the visual embeddings, and the prompt suffix. The text is then generated autoregressively using `lm.generate` with greedy decoding

(`do_sample=False`), which selects at each step the next token with the highest predicted probability, a defined maximum of new tokens, and an optional repetition penalty. The generated token IDs are decoded with the LLaVA tokenizer producing one caption for each input tile, which can then be compared directly with the corresponding ground-truth caption.

4.3.5 Alternative LLM

An alternative code path replaces `LlavaForConditionalGeneration` with `AutoModelForCausalLM` and loads the underlying Vicuna v1.5 checkpoint directly [20]. Since the officially released Vicuna weights are provided only in the legacy `pytorch_model.bin` format, whose loading is incompatible with the versions of `transformers` and `torch` used in this work, the community reupload by Arya [81] is used instead. This version provides the same model weights in the `safetensors` format and avoids the compatibility issue without changing the model itself. The purpose of this variant is to determine whether the observed downstream behaviour depends on the LLaVA wrapper or solely on the frozen Vicuna base model. The training objective, sequence construction, and decoding procedure remain unchanged.

4.4 Bridge

The bridge is a small trainable projection module that maps the frozen TerraMind encoder tokens into the input embedding space of the frozen LLaVA-1.5 decoder. It is the only component of the pipeline that is updated during training. Two architectural variants are implemented and selected using a configuration key. The first uses the linear projection from the original LLaVA architecture [23], while the second uses the two-layer MLP projector introduced in LLaVA-1.5 [21]. The two variants differ only in one aspect, which is whether the mapping from the encoder space to the embedding space is affine or not. This makes the comparison between the two variants particularly useful.

4.4.1 Role and shape

The bridge connects two models that were pretrained independently and were not trained to communicate with each other. The encoder produces one embedding vector for each image patch from its last layer. These vectors are then arranged along a sequence axis according to the order in which the patches cover the scene. With the crop size used in all experiments, this results in $N = 256$ tokens for each modality. Since the optical and SAR data are trained in separate runs (Section 4.1), only one modality is active at a time. Therefore, the bridge receives a tensor with the shape

B $\times$ 256 $\times$ 768, where B represents the batch size, and produces a tensor with the shape B $\times$ 256 $\times$ 4096.

Neither of these dimensions is chosen freely within the method. The encoder dimension $d_{\text{enc}} = 768$ corresponds to the width of the TerraMind base encoder, while $d_{\text{lm}} = 4096$ represents the hidden size of the LLaVA-1.5 language backbone. Both dimensions are therefore determined by the surrounding frozen models. In the implementation, these values are not directly specified in the code. Instead, the input dimension is obtained at build time through a probe forward pass of the frozen encoder, while the output dimension is taken from the configuration of the loaded language model. This allows the same bridge implementation to be used when one of the connected models is replaced by another model with a different width.

The projection is applied separately to each token. The same weights are used for all 256 tokens. As a result, the bridge does not depend on the sequence length N. Using the same weights for all positions is an intentional design choice and not only a technical simplification. If a separate matrix were used for each position, the number of trainable parameters would increase from around 3.15 million to approximately 807 million. This would go against the idea of training a small module between two frozen models. In addition, each matrix would be trained with 256 times fewer examples. A position-specific matrix could also map two patches with the same content to different points in the embedding space simply because they occur at different positions in the scene. However, the position of a patch in the grid does not provide a reason to translate its features differently. Therefore, the same transformation is applied to all positions.

Consequently, the bridge does not perform any interaction between different tokens. Any spatial reasoning required for generating the caption must instead be performed by the causal self-attention layers of the frozen language model. These layers process the projected tokens as a normal prefix of the input sequence. Additionally, the bridge does not introduce any additional positional information. The tokens already contain the positional encoding provided by the encoder, and the projection keeps their original order. Therefore, the language model receives the tokens in exactly the same order as they were produced by the encoder.

The projected visual tokens from the bridge are placed in the language model input sequence between a fixed instruction prefix and suffix. The bridge is not trained with a separate target. Its parameters are updated only through the caption loss described in Section 4.5.

4.4.2 Linear projection

The linear variant applies a Layer Normalisation *LN* [82] over the encoder dimension, followed by a single learnable affine projection

$$v_{\text{linear}} = W\,\text{LN}(x) + b, \qquad W \in \mathbb{R}^{4096\times 768},\ b \in \mathbb{R}^{4096}.$$

The normalisation is applied before the projection because the bridge does not control the magnitude of the encoder tokens. TerraMind was trained with its own objective, so the scale of the tokens produced by its last block is determined by the previous training. Applying normalisation first removes this uncertainty in the input scale and gives the projection a more consistent input. Therefore, the range of the projected tokens at the beginning of training mainly depends on the initialisation of W and can be estimated in advance. The normalisation also gives all tokens a similar norm. As a result, a token does not contribute more to the weight update only because its feature values have a larger magnitude. This is important here because the same architecture is trained with both optical and SAR tokens in the two parallel lines.

The weight matrix W is initialised using Xavier uniform initialisation [83], while the bias b is initialised to zero. The linear variant is not simply a weaker baseline. It allows to investigate how much of the captioning performance can already be achieved when the geometry of the encoder representation is only rotated, scaled, and shifted.

4.4.3 MLP projection

The MLP variant follows the two-layer projector introduced in LLaVA-1.5 [21]. The increase from the encoder width to the language model width takes place completely in the first layer. The second layer then operates within the embedding space of the language model. It also first applies Layer Normalisation to the encoder tokens and then uses a linear layer followed by a GELU activation and a second linear layer

$$v_{\text{mlp}} = W_2\,\text{Drop}\big(\phi(W_1\,\text{LN}(x) + b_1)\big) + b_2,$$

where $W_1 \in \mathbb{R}^{4096\times 768}$, $W_2 \in \mathbb{R}^{4096\times 4096}$, and ϕ = GELU is the activation function introduced in Section 2.1.

The activation function is the main difference between this variant and the linear one. Since ϕ is non-linear, the resulting mapping is no longer affine. Therefore, the token representations are not limited to a 768-dimensional affine subspace, and the module can learn different transformations for different regions of the encoder space. This additional capacity is what the comparison between the two variants is intended to evaluate. However, it also comes with roughly six times as many parameters.

A dropout layer *Drop* [84] is applied after the activation with a rate determined in the configuration file. The layer uses the inverted dropout formulation. During training, each unit is set to zero with probability p, while the remaining activations are scaled by $1/(1-p)$. During evaluation, dropout is disabled and the mapping is therefore deterministic. Dropout is not included in the original LLaVA-1.5 projector. It is added in this work because the bridge is the only trainable part of the pipeline, while the caption dataset used here is several orders of magnitude smaller than the visual instruction data used to train LLaVA-1.5. When $p = 0$, dropout becomes an identity mapping. In this case, the module is exactly the LLaVA-1.5 projector with input normalisation applied before it.

Both linear layers use Xavier-uniform initialisation and zero biases. The random initialisation breaks the symmetry between output neurons, which would otherwise receive identical gradients and remain identical throughout training [2, p. 301].

4.4.4 Implementation and number of parameters

The bridge is selected for each experiment using the `bridge_type` key in the configuration file and applied by the `build_bridge` function. It returns either a `LinearBridge` or an `MLPBridge`, both implemented as standard PyTorch `nn.Module`s [80]. Table 4.1 lists the trainable tensors of both bridge variants. For an encoder dimension of 768 and a language model hidden size of 4096, the linear bridge contains approximately 3.15 million trainable parameters, whereas the MLP bridge contains approximately 19.9 million, resulting in a factor of 6.3 between the two variants. Despite this difference, both bridges remain small compared to the frozen components of the overall pipeline. The frozen LLaVA-1.5 decoder alone contains roughly 7 billion parameters, meaning that the linear bridge accounts for only about 0.04% of this parameter count, while the MLP bridge accounts for about 0.28%.

Table 4.1: Trainable tensors of the two bridge variants for $d_{\text{enc}} = 768$ and $d_{\text{lm}} = 4096$. All remaining parameters of the pipeline are frozen.

Tensor	Shape	Linear	MLP
W_1	4096×768	3 145 728	3 145 728
b_1	4096	4 096	4 096
W_2	4096×4096	–	16 777 216
b_2	4096	–	4 096
Total		3 149 824	19 931 136

4.5 Training Objective and Protocol

Each training step includes a forward pass through the frozen encoder within `torch.no_grad`, followed by the trainable bridge and the loss computation. The loss

is then backpropagated through the frozen language model, with gradients flowing back only to the bridge.

4.5.1 Loss

The training loss is the next-token cross-entropy calculated by the frozen language model over the caption positions of the sequence described in Section 4.3. Since the reference caption is part of this sequence, each caption token is predicted using the reference tokens that come before it instead of the model's own previous predictions. This corresponds to the teacher-forcing approach [2, pp. 381-384].

As described in Section 4.3, the visual positions do not have target labels. They are therefore not predicted directly and do not contribute a separate term to the loss. In a causal transformer, information at these positions can affect the loss through the later positions that attend to them [3]. The attention mask allows every caption position to attend to the visual prefix, so the projected tokens can influence the loss through the attention paid to them by the caption tokens.

The same loss is used for evaluation on the validation split. For this evaluation, the bridge is set to `eval()` mode, which disables dropout, and each validation scene is paired with one fixed reference caption. This allows the training and validation losses to be compared on the same scale. The bridge is not trained with a contrastive loss, a reconstruction loss, or an explicit alignment loss. Its only training signal comes from increasing the probability of the reference caption while the parameters of the language model remain frozen.

4.5.2 Optimizer

The bridge parameters are optimised using AdamW [85] from the PyTorch implementation [80]. AdamW applies weight decay separately from the gradient update, so the regularisation does not directly affect how the gradients are used to update the parameters. It is also the optimiser used in both training stages of LLaVA-1.5 [21]. A weight decay, defined in the configuration file, is applied together with the default AdamW parameters. Before each optimiser step, gradients are clipped to a global L_2-norm of 1.0 to prevent occasional large gradients, for example from longer captions, from destabilising training. This also matches the value applied in the original LLaVA-1.5 pipeline.

4.5.3 Learning-rate schedule

The learning rate follows a two-stage schedule implemented with `SequentialLR`. First, it is linearly increased from 1% of the target learning rate to the target value

during the first 3% of the total optimiser steps. This warmup provides a more stable start for the AdamW optimiser.

Afterwards, the learning rate is reduced using cosine annealing [86], decreasing from the target value to a set minimum over the remaining training steps. The schedule is updated after each optimiser step, so the learning rate depends on the training steps and not directly on the epochs. The total number of optimiser steps is determined by the configured number of epochs and the number of batches in each epoch for the selected batch size. Since incomplete final batches are discarded, the number of steps per epoch is given by the number of training samples divided by the batch size and rounded down.

4.5.4 Diagnostic measures

The target learning rate is not predefined by the method and was determined empirically using a sweep over a logarithmically scaled interval. The value resulting in the lowest validation loss was selected. The procedure is described in Section 5.1.3, while the remaining hyperparameters are listed in Table 5.1.

Together with the training and validation losses, an additional measure is recorded in TensorBoard. The *grounding gap* on the validation split compares the loss for the correct image-caption pairs with the loss after the visual inputs have been permuted within each batch. A shift with a fixed offset is used for the permutation, so each caption is paired with the visual tokens of another scene while the procedure remains the same across epochs. The grounding gap is defined as the loss of the permuted pass minus the loss of the correct pass. A positive value therefore indicates a higher loss when the correspondence between the image and caption is removed. A larger grounding gap suggests that the bridge output contains scene-specific information that is relevant for the caption prediction.

The grounding gap only considers the loss and does not directly measure the quality of the generated captions. It is therefore used together with the reference conditions described in Section 5.2.2. The grounding gap has a limited role in the evaluation. Its curves are inspected manually to check whether the projected visual tokens have a measurable influence on the caption loss. No threshold is used for this analysis, and the measure has no influence on the learning rate, training duration, or checkpoint selection.

4.6 Design Choices

This section elaborates the most important design choices and outlines the alternatives that were considered.

4.6.1 Encoder choice

The central research question requires a vision backbone that incorporates both optical and SAR data during pretraining, as otherwise the experiments would lack a meaningful vision-language foundation. The main model families were considered in Section 2.5, contrastive dual-encoder models adapted to remote sensing, including RemoteCLIP, and several SAR-CLIP variants, and the cross-modal reconstruction approach represented by TerraMind.

The contrastive models were not selected because none of the considered variants is pretrained on SAR from the beginning. Instead, they start from a natural-image CLIP backbone and are subsequently adapted to remote sensing. Even the SAR-CLIP variants pass through an optical remote-sensing stage before SAR training. Their representations are therefore primarily shaped by optical imagery. In addition, the associated SAR-caption pairs are either derived from accompanying optical descriptions or limited to predefined templates, making the language supervision only indirectly related to the SAR observations. TerraMind was chosen instead as a state-of-the-art publicly available encoder that is pretrained jointly and from scratch on Sentinel-1, Sentinel-2, and captions using cross-modal reconstruction. This results in a shared representation space in which language information can influence the representation of different modalities, including SAR.

Within the TerraMind family, the base variant was selected over the larger 1024-dimensional model due to the available compute budget. A limitation of this choice is that TerraMind is not explicitly language-aligned in the same way as contrastive models such as CLIP, placing more of the vision-language mapping burden on the trainable bridge. Furthermore, the pretraining captions are based on RGB descriptions [38], so SAR-specific concepts and vocabulary are not directly supervised and therefore have to be learned or emphasised during downstream training.

4.6.2 Bridge-only training without LoRA

The frozen-encoder and frozen-decoder setup limits the trainable parameters to the bridge. Full fine-tuning of the 7B-parameter LLaVA-1.5 decoder was ruled out because the required memory for gradients and optimiser states exceeds the available per-job memory. LoRA [16] would be feasible within the compute budget, but was not used because it introduces additional hyperparameters, which would require further experimenting and would exceed the scope of this thesis.

Bridge-only training avoids these additional factors and keeps the number of trainable parameters relatively small. It also isolates the contribution of the TerraMind representations to downstream captioning and fits well within the available compute budget. The main trade-off is that captioning performance may be lower than with

approaches that also fine-tune the language model. However, the aim of this work is primarily to determine whether TerraMind's cross-modal representations contain sufficient information for language generation, particularly when only a single modality is available at inference time. More extensive fine-tuning is therefore left for future work.

5 Experiments

Several experiments have been performed with different variants. This chapter describes those variations, the captions generated by the resulting models, and their comparison with an external baseline that does not use a geospatial encoder.

Before the results for the optical and SAR tracks are presented separately, the evaluation metrics are introduced. The analysis also examines the contribution of the true visual input on the measured performance. It concludes with a discussion of the main findings and their implications.

5.1 Experimental Setup

The architecture, sequence assembly, loss function, and optimisation procedure follow Chapter 4. The two captioning tracks in Section 5.4 and Section 5.5 apply the same design to optical and SAR supervision, respectively. The setup is therefore described once here and applies to both tracks.

5.1.1 Data and Evaluation Splits

All runs use the SSL4EO-S12 v1.1 archive described in Chapter 3. The split into training, validation, and test data is performed at shard level. This ensures that no scene contributes to more than one split and that no two acquisitions from the same location are distributed across different splits. The training split comprises 3,840 scenes, the validation split comprises 640 scenes, and the test split contains 1,280 scenes.

Each scene contains four seasonal acquisitions, with one caption assigned to each acquisition. Acquisitions are treated as separate samples throughout the experiments because each caption describes the state of the surface at the time of acquisition, rather than the location itself. The test split therefore provides $1{,}280 \times 4 = 5{,}120$ image–caption pairs. This presents the sample size underlying all test-set results reported in this chapter.

The two tracks use the two caption sources described in Section 3.3. Each track is evaluated against its corresponding reference dataset. The SAR references consist of free-form prose with an average length of 88 words, whereas the optical references are Question-Answering-based and have an average length of 44 words. Consequently, absolute metric values are not directly comparable across the two tracks. Within each

track, all conditions are evaluated against the same references and can be compared directly.

5.1.2 Training Protocol and Fixed Configuration

The parameter values used for the experiments are summarised in Table 5.1. They cover the settings that can be changed through the configuration of the components described in Section 4.3 to Section 4.5, but remain the same for all experiments in this chapter. Settings that are relevant to the baseline are also kept the same where this is possible.

Table 5.1: Configuration values held constant across all runs.

	Parameter	Value
Data	Input crop	256×256
	Acquisitions per scene	4
Optimisation	Batch size	16
	Maximum epochs	10
	Early-stopping rule	5% above minimum
	Weight decay	1×10^{-4}
	Bridge dropout	0.1
Decoding	Maximum new tokens	200 (S1), 100 (S2)
	Repetition penalty	1.2
Randomness	Seed	0

All conditions are trained only once using this seed. No additional runs with other seeds were performed, so the results do not provide an estimate of the variation caused by different initialisations or data orders.

For each epoch, the complete training split is processed once before the model is evaluated on the validation data. The validation stage provides both the validation loss and the grounding gap described in Section 4.5.4. The model state with the lowest validation loss is selected as the final checkpoint for the run. Training is stopped when the validation loss has risen over five percent of the minimum.

Caption generation uses greedy decoding for all experimental conditions. The next token is therefore always selected from the model output without any random sampling. Given the same input image and model parameters, the generated caption is consequently deterministic. Repeating the evaluation with the same checkpoint and input produces the same caption, so differences between conditions are not caused by sampling variation.

The maximum number of generated tokens and the repetition penalty are kept constant for all conditions within a track. The same settings are also used for the baseline

described in Section 5.2.3. Keeping these parameters fixed matters because the evaluation metrics react differently to changes in caption length. Using the same decoding settings prevents this from becoming an additional source of variation between the experimental conditions.

The maximum token budget is different for the two tracks because their reference captions have different lengths, as described in Section 5.1.1. In both cases, the selected limit is sufficiently high to generate a complete reference caption.

Hardware and wall-clock time per run are documented in Chapter C.

5.1.3 Selection of the Learning Rate

The peak learning rate was the only hyperparameter selected through an iterative search. The remaining values in Table 5.1 were fixed beforehand based on a series of preliminary runs. For these runs, one setting was changed at a time while the rest of the configuration remained unchanged, and the results were compared using the validation loss. The grounding gap described in Section 4.5.4 was monitored as an additional measure to see whether a change influenced the model's reliance on the visual input.

A batch size larger than 16 was not possible because it exceeded the available GPU memory. The size of the training split was also tested with different values. Larger training sets resulted in lower validation losses, so the final split uses the largest dataset that could be trained within an applicable time limit. Weight decay and bridge dropout were both tested with several values, but neither the validation loss nor the grounding gap showed a noticeable change. The values listed in Table 5.1 were therefore kept for the final experiments. Since these preliminary runs were not part of a systematic hyperparameter search, their individual results are not reported in detail.

The learning rate directly controls the size of the parameter updates of the bridge and has a relatively large possible range compared to the other optimisation settings. The search was performed with the Optuna framework [87], which was integrated into the training procedure. A trial differs from a normal training run only in that the selected parameters are written to the configuration before training and the validation loss is reported after each epoch.

The search space is log-uniform and covers $[1\times10^{-5}, 3\times10^{-4}]$ for the MLP bridge and $[3\times10^{-5}, 1\times10^{-3}]$ for the linear bridge. This allows learning rates to be tested across several orders of magnitude. The candidate values are proposed by a Tree-structured Parzen Estimator [88] using a fixed seed, so the sequence of trials can be reproduced. Each study contains twelve trials. The minimum validation loss reached during a trial

is used as the objective value and also determines the checkpoint selected for the main experiments.

Two pruning rules are used to stop trials early when their results are not promising. One compares a trial with the other trials in the same study, while the other compares it with the best result reached by the trial itself. The parameters of both rules are described in Section B.3.

To make the selection reproducible, the sweep configuration, study database, resolved configuration of each trial, and the selected trial are stored and can be inspected without repeating the search, as described in Section 5.1.4.

5.1.4 Reproducibility

Each training run is defined by a single YAML configuration file. The entry points use a `--config` argument and do not accept individual hyperparameters from the command line. Changes to the configuration therefore have to be made in the configuration file, which is version-controlled and can be referenced later.

Before training starts, the complete configuration is saved to the run directory as `config.yaml`. This also includes values that were taken from the default configuration. The settings of a run can therefore be recovered directly from its output directory. During inference, this configuration is used to reconstruct the encoder, bridge, and decoder, ensuring that the checkpoint is evaluated with the architecture specified for the corresponding run.

Run directories are also generated from the configuration. The directory name contains the bridge type, modality, peak and minimum learning rate, an optional modality-dropout marker, and the launch timestamp. This states the main experimental factors directly in the directory structure and prevents runs with the same configuration from overwriting each other.

Figure 5.1 shows the files generated by a completed training run and a hyperparameter sweep. A training run stores the resolved configuration, the checkpoints, and the TensorBoard event files. The generated captions are added during inference. A sweep additionally stores its configuration, the Optuna study as a SQLite database, the configuration of each trial, and the best trial. This allows the selection of the peak learning rate to be inspected without running the complete sweep again.

5.2 Conditions under Comparison

The experimental conditions cover three different variants of the proposed system, a case where the visual input does not correspond to the reference caption, and an

```
experiments/
├── mlp_s1_0.0001_1e-05_26-08-16_12-55/
│   ├── config.yaml
│   ├── checkpoints/
│   │   ├── best.pt
│   │   ├── epoch_0005.pt
│   │   └── ...
│   ├── runs/
│   └── captions/
│       ├── batch_000021_S1.csv
│       └── ...
└── bridge_mlp_s1_v1/
    ├── sweep_config.yaml
    ├── optuna.db
    ├── best_trial.yaml
    ├── trial_0000/
    │   └── config.yaml
    └── ...
```

Figure 5.1: Artefacts of a training run (top) and a hyperparameter sweep (bottom). Both directory names are generated from the configuration.

external baseline using a different visual processing pipeline. An overview of all conditions is provided in Section 5.2.4.

5.2.1 Design Factors

The reference setup uses the MLP bridge together with the LLaVA-1.5 decoder and the default instruction introduced in Section 4.3. The experimental conditions are constructed by changing one factor at a time, which allows the effect of each component to be evaluated separately. An overview of the resulting conditions is given in Table 5.2. The implementation and function of the individual variants are described in Section 4.4 and Section 4.3.

Condition D1 examines whether the additional layer in the MLP bridge improves the alignment between the encoder and decoder representations. In both variants, the 768-dimensional encoder tokens are first projected into the decoder's 4,096-dimensional embedding space. The linear bridge ends after this projection, whereas the MLP applies a GELU activation and performs another transformation within the same dimensionality. The comparison therefore focuses on the effect of this second, non-linear transformation. The MLP adds 16.7 million trainable parameters to give the bridge more freedom when transforming the encoder features.

Table 5.2: Design factors. Each condition differs from the base configuration in exactly one component. Deviations from the base configuration are shown in bold.

	Bridge	Decoder	Prompt	Question addressed
Base	MLP	LLaVA-1.5	default	—
D1	**linear**	LLaVA-1.5	default	Does the bridge benefit from non-linearity?
D2	MLP	**Vicuna v1.5**	default	Does visual instruction tuning of the decoder matter?
D3	MLP	LLaVA-1.5	**sensor-aware**	Can sensor context be supplied through the prompt?

Condition D2 compares the LLaVA-1.5 decoder with Vicuna v1.5, based on the two decoder paths described in Section 4.3. Since LLaVA-1.5 is built on Vicuna v1.5, the two models use the same underlying architecture, tokeniser, and pretraining dataset. The visual components of the original LLaVA wrapper are removed in both cases. The main difference is that LLaVA-1.5 was further trained with visual instruction data, during which the language-model weights were also updated [21]. This comparison therefore examines whether this additional training improves the frozen decoder's ability to interpret the visual representations provided by the bridge, beyond its general language modelling capabilities.

Condition D3 examines a property specific to the proposed pipeline and is evaluated only for the SAR track. The visual tokens do not contain explicit information about the sensor that produced them. This is particularly relevant for SAR data, where the decoder is expected to describe backscatter phenomena that were not part of its original training. The sensor-aware prompt introduces a remote-sensing role in the prefix and asks for descriptions based on observable scattering signatures in the suffix. The tested prompt is given in Section A.2. Since the instruction is the only component that differs from the default condition, the comparison shows how much can be provided through the language input and how much has to be learned through the visual alignment.

5.2.2 Control Condition

The design factors described above only compare different versions of the proposed system. They do not by themselves show whether the generated captions contain information about the corresponding scene, since all conditions use correctly paired visual inputs. In addition, the decoder may already have learned common vocabulary and descriptions that occur frequently in the reference corpus. To examine the contribution of the scene-specific information, an additional control condition is therefore introduced. The generated captions are paired with the reference captions of different scenes for evaluation. The trained bridge, decoder, prompt, decoding

configuration, and reference corpus remain unchanged. The condition therefore differs from the base configuration only in the pairing used for the metric calculation. It is referred to as the *shifted-captions* condition in the following. The difference between the scores of the base configuration and this condition, denoted by Δ in Section 5.6, indicates how much the evaluation score changes when the generated caption is compared with the reference caption belonging to the same scene. It is used as the text-side measure of visual grounding.

The resulting score should not be interpreted as a chance level. This shift can still match similar scenes and therefore reach a higher score than a completely different scene would reach. The shifted-captions score therefore represents a reference point for the score that can be obtained without the correct scene. It is not a lower bound corresponding to random performance. The retrieval experiment in Section 5.6.2 provides an additional measure for which a defined chance level is available.

The grounding gap introduced in Section 4.5 is based on the same general principle. During training, visual inputs are permuted within a batch and the resulting change in validation loss is recorded. The shifted-captions condition applies this idea during evaluation and measures the corresponding change in caption scores. The two measures therefore capture visual dependence at different stages of the pipeline. The loss-side gap indicates whether the training objective is affected by replacing the correct visual input, while Δ indicates whether this effect is also visible in the generated captions.

5.2.3 External Baseline

The baseline uses the complete LLaVA-1.5 checkpoint [21], including the vision tower and projector that are removed from the bridge-based setup described in Section 4.3. Since this is the same checkpoint that provides the decoder in the base configuration, the language-model weights are identical in both cases. The baseline is evaluated without any additional training and without an adapter, making it a zero-shot condition. The remaining evaluation setup is kept unchanged. The instruction is taken from Section 4.3. For SAR, this also includes the sensor-aware prefix from condition D3 and an additional sentence specifying the colour assignment of the polarisation channels. This information is required because the bridge receives the two polarisation channels as separate inputs, while the baseline receives a three-channel colour image. The prompts are listed in Section A.3.

The baseline images are generated from the same archives as the inputs used by the bridge experiments. Both approaches therefore use the same acquisition and centre crop. For the optical track, bands B04, B03, and B02 are combined into a true-colour image, and for the SAR track, the two polarisation channels are converted into a false-colour image as already described in Section 3.3.2.

The baseline provides an external reference point and is not designed as a controlled comparison of the encoder alone. For the optical track, the baseline receives three of the twelve bands available to the geospatial encoder. For SAR, the two polarisation channels are converted into three colour channels before they are processed by the vision tower. Both inputs also undergo a non-linear transformation and quantisation to 256 levels per channel, while the geospatial encoder processes the original values at full precision. A performance difference therefore represents the effect of the complete processing pathways. One pathway uses the original measurements, the geospatial encoder, and the trained bridge, while the other uses the rendered image, the vision tower, and the pretrained projector.

5.2.4 Summary of Conditions

Table 5.3 lists every condition that appears in the results tables of this chapter.

Table 5.3: All conditions evaluated in this chapter. The design factors differ from the base configuration in exactly one component. The two points of comparison are not variants of the proposed system. The shifted-captions condition withholds the correct scene from it, and the baseline replaces its visual pathway entirely.

	Visual pathway	Bridge	Decoder	Prompt	Status
Proposed system					
Base	TerraMind	MLP	LLaVA-1.5	default	trained
D1	TerraMind	linear	LLaVA-1.5	default	trained
D2	TerraMind	MLP	Vicuna v1.5	default	trained
D3	TerraMind	MLP	LLaVA-1.5	sensor-aware	trained, SAR only
Points of comparison					
Shifted captions	TerraMind	MLP	LLaVA-1.5	default	trained
Baseline	CLIP vision tower	pretrained projector	LLaVA-1.5	default (S2), sensor-aware (S1)	zero-shot

5.3 Evaluation Metrics

Each generated caption is compared with the reference caption belonging to the same acquisition. Four metrics are used for this evaluation, as they measure different types of agreement between the generated and reference captions. BLEU-4 and ROUGE-L are based on word overlap, while METEOR also considers a limited amount of lexical variation through stemming and synonyms. The SBERT cosine similarity compares the corresponding sentence embeddings instead of individual words. Using all four metrics therefore provides a broader view of the generated captions. Improvements across all metrics indicate better agreement in both wording and meaning, while differences between the metrics can show which aspect has changed.

All metrics are calculated separately for each sample and stored for all 5,120 image-caption pairs in the test split. The results can therefore be compared pairwise on a sample level.

5.3.1 Definition of the Metrics

BLEU-4. BLEU measures how many word sequences in the generated caption also occur in the reference caption. For each n-gram, the number of matches that are counted is limited to the number of times that n-gram occurs in the reference. Without this limit, a caption could repeat a single n-gram that appears in the reference and still reach a high precision, since every repetition would be counted as a match. A repetition therefore only contributes to the score if the reference repeats the n-gram as well. The resulting precision values for n from one to four are combined using a geometric mean. A brevity penalty is then applied to reduce the score of captions that are shorter than the reference. The score ranges from 0 to 1, but a value close to 1 is uncommon unless the generated caption is very similar or identical to the reference. A caption can therefore describe the scene correctly and still receive a relatively low score if it uses different wording. [89]

BLEU was originally defined for complete test datasets, and the authors point out that scores for individual sentences can differ from human judgement [89]. When BLEU is calculated for a single caption, the score also becomes zero if there is no matching four-gram, even when shorter n-grams match. A smoothing method is therefore required for sentence-level evaluation [90]. In this work, a zero match count is replaced by a small positive constant. The resulting values are therefore sentence-level smoothed BLEU scores. They cannot be directly compared with dataset-level BLEU scores from other studies. BLEU scores also require identical tokenisation and the same metric variant to be comparable [91].

ROUGE-L. ROUGE-L is based on the longest common subsequence of the generated and reference captions. The words in the subsequence do not have to be adjacent, so the metric can reward content that appears in the correct order even when other words occur between the matching words. The length of the common subsequence is divided by the length of the reference to obtain recall and by the length of the generated caption to obtain precision. These two values are then combined into an F-measure. The score is 1 if the two captions are identical and 0 if they have no word in common. Unlike BLEU, ROUGE-L does not depend on consecutive n-gram matches or on a fixed n-gram length. [92] The implementation used in this work gives equal weight to precision and recall [93]. A caption can therefore receive a lower score both when it leaves out content from the reference and when it adds content that is not present in the reference.

METEOR. METEOR aligns the words of the generated and reference captions, with each word being matched to at most one word of the other caption [94]. Identical word forms are matched first, followed by words with the same stem and finally by words that share a sense in WordNet [95]. Precision and recall of the resulting matches are combined into a harmonic mean, with recall receiving nine times the weight of precision. A fragmentation penalty is then applied based on how much the matched words are distributed across separate parts of the caption. The penalty can reduce the score by up to one half. The METEOR score ranges from 0 to 1, with higher values indicating greater agreement. METEOR was designed for evaluation at the level of individual segments, which is also how it is used in this work. [94] The synonym matching has limited importance for the data used here, since WordNet covers general English while the SAR references from Section 5.1.1 contain terms describing scattering behaviour, such as double-bounce and backscatter.

SBERT cosine. The three metrics above compare the generated and reference captions mainly at the word level. A caption can describe the same scene with different wording and still receive a low score. The fourth metric therefore compares sentence embeddings. SBERT fine-tunes a pretrained transformer so that sentences with similar meanings have similar representations in the resulting vector space. The sentence embeddings can then be compared using their cosine similarity. As a cosine similarity, the score lies between -1 and 1, with higher values indicating more similar sentence embeddings. [96] The model used in this work is `all-mpnet-base-v2` [97].

The cosine similarity does not have a meaningful zero point for this application because even unrelated English sentences about landscapes can share vocabulary and sentence structure. The measured values therefore occupy a relatively narrow range in the upper part of the scale. An individual value is consequently difficult to interpret on its own and is instead considered in relation to the reference condition of Section 5.2.2. The model was also fine-tuned on general-domain text, and its authors report lower performance for topics that are outside the training data. This is relevant for the present application because the captions describe Earth observation scenes and contain domain-specific terms.

5.3.2 Implementation and Reported Quantities

BLEU-4 and METEOR are calculated using the implementations provided by the NLTK library [98]. ROUGE-L is calculated with the HuggingFace `evaluate` library [93], which uses the reference implementation of `rouge_score`. The SBERT cosine similarity is calculated with the `sentence-transformers` library. For BLEU and METEOR, the captions are converted to lower case and tokenised with the NLTK word tokeniser. Punctuation marks are therefore treated as separate tokens. The ROUGE

implementation uses its own tokenisation and removes punctuation, so the metrics are calculated according to the conventions of their respective implementations. Since the same procedure is used for all experimental conditions, this difference does not affect the comparison between them.

Each acquisition has exactly one reference caption, as described in Section 5.1.1. The metrics were originally developed for settings where several reference captions can be available for one sample. Multiple references make it possible to give credit to different valid formulations when they occur in one of the references. BLEU can also be used with a single reference for a sufficiently large test dataset, provided that the references are not all produced from the same source [89]. This condition is not fully satisfied in the present dataset because the references within each track were generated by the same process and follow a relatively uniform style. The absolute metric values are therefore influenced by the limited reference diversity. This affects all experimental conditions within a track and is therefore mainly relevant when interpreting the absolute values, while comparisons between conditions remain possible.

For each condition, the reported metric value is the mean over the 5,120 test samples together with a 95% confidence interval calculated using the normal approximation. The interval describes the variation across the test samples within a single run. For the shifted-captions condition from Section 5.2.2, the difference Δ from the base configuration is also reported. This difference is especially useful for metrics without a meaningful zero point because it directly shows the change in score between the correct and mismatched caption pairings.

Because the test split contains 5,120 samples, the confidence intervals are narrow. A difference between two conditions can therefore be clearly separated and still be small in relation to the other differences reported in this chapter. Consequently, for every comparison is stated, whether the intervals are separated, and how large the difference is compared with the distance between the trained conditions and the zero-shot baseline on the same metric.

5.3.3 Retrieval Measures

The four metrics introduced above evaluate each generated caption against the single reference caption from the same acquisition and produce one score for the pair. Such a score can also be high for a caption that is fluent and representative of the dataset but contains little information about the specific scene. Hodosh, Young, and Hockenmaier [99] address this limitation by formulating image annotation as a ranking problem over a set of candidate captions. They show that metrics based on the ranking of multiple results can provide a more robust evaluation than measures based on one generated response per query. The same measures are used for the additional check described in Section 5.6.2.

Recall at position k, denoted by R@k, measures the fraction of queries for which the correct item occurs within the first k results. Higher values of R@k and a lower median rank indicate that the correct item is ranked closer to the top. It is reported for $k \in 1, 5, 10$. R@50 is also taken into account because the galleries used in section 5.6.2 are larger than those in the cited work. The median rank describes the rank at which half of the queries have found their correct item. Precision is omitted because every query has exactly one correct result. [99] The same values of k are also commonly used for cross-modal retrieval evaluation in remote sensing [27, 36].

The retrieval setup used here is slightly different from the one in [99]. In their work, an image is used as the query and a set of existing captions is ranked. Here, the caption is generated first, and the ranking is performed afterwards as an additional evaluation. The ranking is also done between captions. The generated caption is used as the query, while the reference captions of the corresponding track are used as candidates. There are N = 5,120 test pairs in total. The N reference captions are embedded with the SBERT model described in Section 5.3.1, and for each generated caption the candidates are sorted according to their cosine similarity to it. The rank r is the position of the correct candidate in this ranking, where r = 1 means that it is the most similar one. Candidates are shuffled with a fixed seed before sorting, so that a query whose similarities are all equal receives a uniformly random rank instead of rank one.

What counts as the correct candidate is defined at two levels. At caption level, the candidate pool contains all N reference captions of the track, and a query is correct only when the reference caption of its own acquisition is ranked first. Each scene contributes four seasonal acquisitions that describe the same location, as described in Section 5.1.1, so the references of the three remaining acquisitions of the same scene are part of this pool and can be very similar to the correct one. At scene level, it contains the M = 1,280 test scenes instead. Each scene is represented by its four reference captions, and its similarity to a generated caption is the highest of these four values. A query is correct when the scene the caption was generated from is retrieved, regardless of which of its acquisitions provided the best match. The two levels therefore separate whether a caption identifies the location from whether it also identifies the individual acquisition. R@k and the median rank are calculated from the resulting ranks across all N samples at both levels.

This retrieval evaluation also provides a defined reference point for captions that contain no scene-specific information. If the ranking is independent of the generated caption, every position in the gallery is equally likely for the correct candidate. At caption level the expected R@k is therefore k/N and the expected median rank is $(N+1)/2 = 2{,}560.5$. At scene level the corresponding values are k/M and $(M+1)/2 = 640.5$. These values provide a reference for random performance, which is not available for the SBERT cosine similarity itself. They are reported together with the

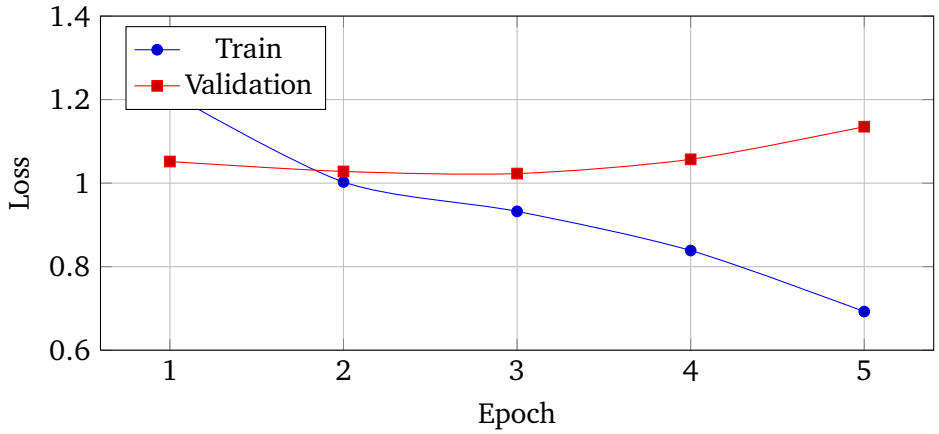


Figure 5.2: Training and validation loss per epoch for the winner trial of the optical track.

measured retrieval results in Section 5.6.2.

5.4 Optical Captioning Results

This section presents the results for the optical track. The bridge is trained using Sentinel-2 input and the SSL4EO-S12 captions as supervision. The experimental conditions are defined in Section 5.2, and all of them are evaluated on the same 5,120 test pairs described in Section 5.1.1.

5.4.1 Training Behaviour

The lowest validation loss of 1.02 was reached at epoch three, as visible in Figure 5.2, so the checkpoint from this epoch was selected for the optical run. Although the configuration allows up to ten epochs, training stopped after epoch five due to the early-stopping criterion described in Section 5.1.2. At this point, the validation loss had increased to 1.14, which was more than five percent above the best value reached so far.

The training and validation losses develop similarly during the first two epochs and then start to diverge. After epoch three, the training loss continues to decrease from 0.93 to 0.69, while the validation loss increases in every following epoch. As a result, the difference between the two losses grows from 0.09 to 0.45. The checkpoint selected at epoch three therefore represents the last point before the model started to improve on the training data while performing worse on the validation data.

5.4.2 Generated Captions

Figure 5.3 shows the three example images from Section 3.3 and their corresponding captions, but this time generated from the developed model in the base configuration in the Sentinel-2 track. The scenes were selected before generating the captions, so the selection was not influenced by the resulting outputs. More examples of the over 5,000 test results can be found in Chapter D.

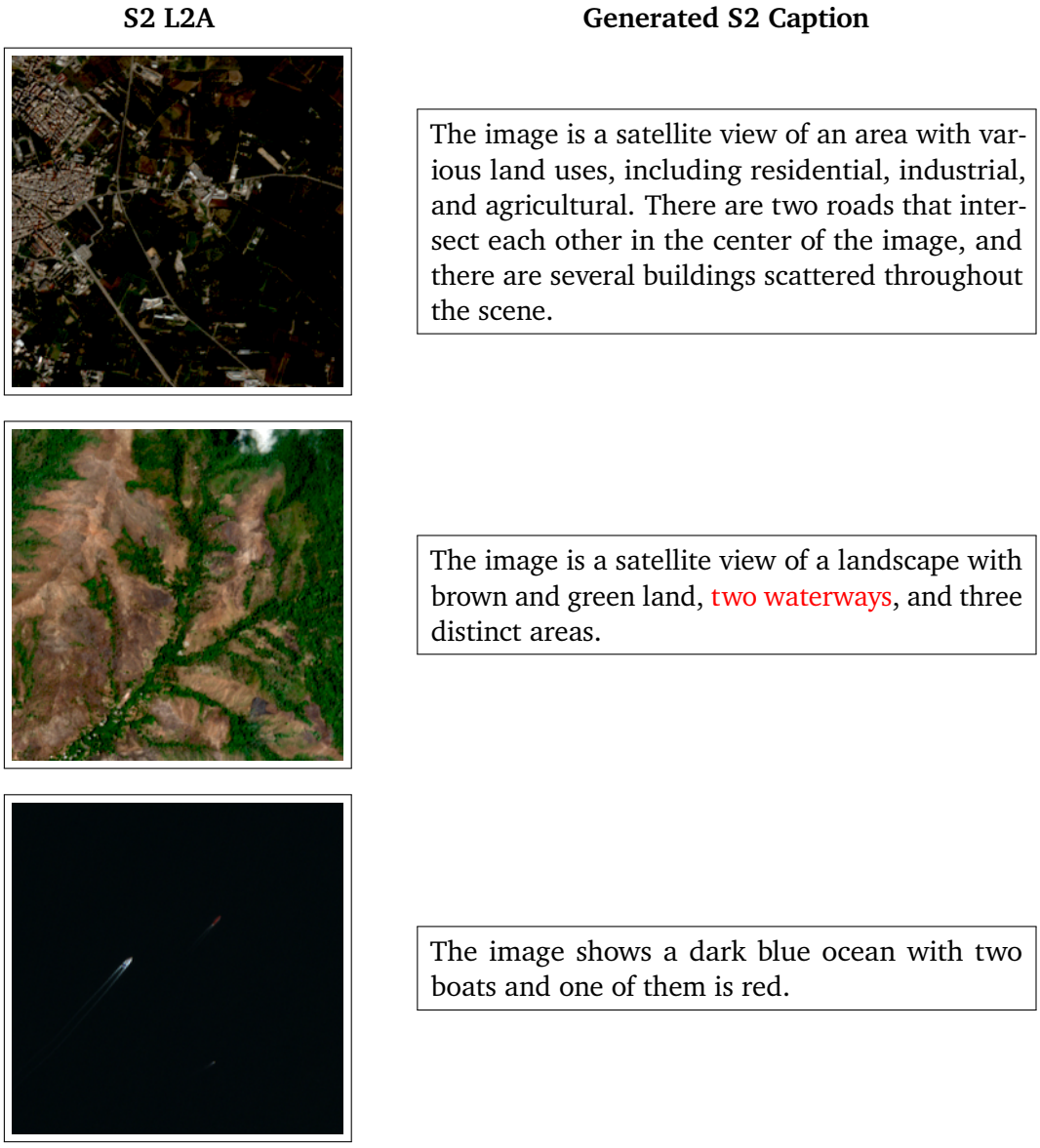


Figure 5.3: Examples of the generated S2 captions. Sentinel-2 L2A is shown as true-color RGB and the upper two images have been brightened for better visibility, just for this presentation. Clearly wrong statements have been marked red. Adapted from [41].

The three captions have a similar level of detail. They mainly describe land-cover classes and common objects, but usually do not include properties that would distinguish one scene from another. The vocabulary is also closely related to the reference dataset, and many of the statements could apply to a large number of the 1,280 test

scenes.

In the first sample, the caption identifies an area with mixed land use containing residential, industrial, and agricultural areas. It also mentions two intersecting roads and several scattered buildings. This general description is correct, since the scene is clearly developed and the mentioned intersection is a visible linear feature. Several details are missing, however. Additional roads cross the tile, and the stadium in the upper left is not mentioned, although stadiums occur consistently in the training captions.

The second example contains an incorrect statement. The caption correctly identifies brown and green areas, but it also mentions two waterways that are not visible in the scene. The brown areas form a branching structure that resembles a drainage network. The remaining description consists of three distinct areas without further details and is therefore difficult to verify from the image. A statement of this type can be neither clearly confirmed nor rejected based on the scene alone.

The third sample is described in more detail. The caption correctly identifies two boats and also mentions the colour of one of them. These are specific objects that are not present in the first two descriptions. The scene itself is comparatively simple and consists mainly of a homogeneous dark surface with a small number of bright objects that stand out clearly from their surroundings.

Section 5.6.2 examines the specificity of the captions for the whole test split.

5.4.3 Metric Results

Table 5.4 reports the metrics of Section 5.3 for all conditions of the optical track.

Table 5.4: Evaluation results of the optical track. Mean over the 5,120 test pairs with the half-width of the 95% confidence interval. The conditions are defined in Table 5.3.

	BLEU-4	ROUGE-L	METEOR	SBERT
Base (MLP)	.1507 ± .0025	.3631 ± .0025	.3210 ± .0034	.7818 ± .0029
D1 (linear)	.1421 ± .0023	.3501 ± .0025	.3173 ± .0031	.7711 ± .0028
D2 (Vicuna)	.1538 ± .0025	.3623 ± .0024	.3299 ± .0031	.7873 ± .0028
shifted-captions	.1274 ± .0022	.3295 ± .0021	.2873 ± .0027	.7100 ± .0035
Baseline	.0260 ± .0006	.1943 ± .0011	.2476 ± .0017	.6262 ± .0026

All three trained conditions achieve higher scores than the zero-shot baseline for all four metrics, and the corresponding confidence intervals do not overlap in any of these comparisons. The size of the difference varies between the metrics. For the base condition, BLEU-4 is 5.8 times higher than the baseline, while ROUGE-L is 1.9 times higher. METEOR shows a smaller increase from 0.2476 to 0.3210. No ratio is given

for the SBERT cosine because the metric has no meaningful zero point, as discussed in Section 5.3.1.

Comparing the linear bridge to the MLP bridge, the base condition obtains higher scores on BLEU-4, ROUGE-L, and SBERT, and the confidence intervals are separate for all three metrics. For METEOR, the intervals overlap and the difference of 0.0037 is therefore not clearly resolved. The largest separated difference is 0.0130 for ROUGE-L. This is small in comparison with the same distance measured against the zero-shot baseline, which is 0.1688 for ROUGE-L, so the second bridge layer accounts for less than a tenth of what the training as a whole contributes on this metric.

When replacing LLaVA-1.5 by Vicuna v1.5 as the decoder, the results of the two conditions are very similar across all four metrics, and their confidence intervals overlap for BLEU-4, ROUGE-L, and SBERT. The only clearly resolved difference is 0.0089 for METEOR and favours Vicuna v1.5. Under these experimental conditions, the visual instruction tuning of LLaVA-1.5 therefore does not provide a relevant improvement.

The shifted condition is included as an additional reference point and does not represent another variant of the proposed system. Its scores are lower than those of the base condition for all four metrics, with non-overlapping confidence intervals, and they remain clearly above the zero-shot baseline. Its BLEU-4 score is 0.1274 compared with 0.0260 for the baseline, and its SBERT cosine is 0.7100 compared with 0.6262. This condition is examined together with the SAR results in Section 5.6.

5.5 SAR Captioning Results

This section presents the results for the SAR track, which trained the bridge with the Sentinel-1 captions described in Section 3.3. The results are presented in the same way as for the optical track in Section 5.4.

5.5.1 Training Behaviour

The lowest validation loss in the SAR run was 1.00, reached at epoch four. The checkpoint from this epoch was selected for the inference and final evaluation. Training stopped after epoch seven, where the validation loss had increased to 1.09. Epoch six was still below this limit by only 0.0004.

The training and validation losses followed each other until epoch four and therefore remained aligned for longer than in the optical track. From epoch five onwards, the training loss continued to decrease from 0.86 to 0.70, while the validation loss increased in each epoch.

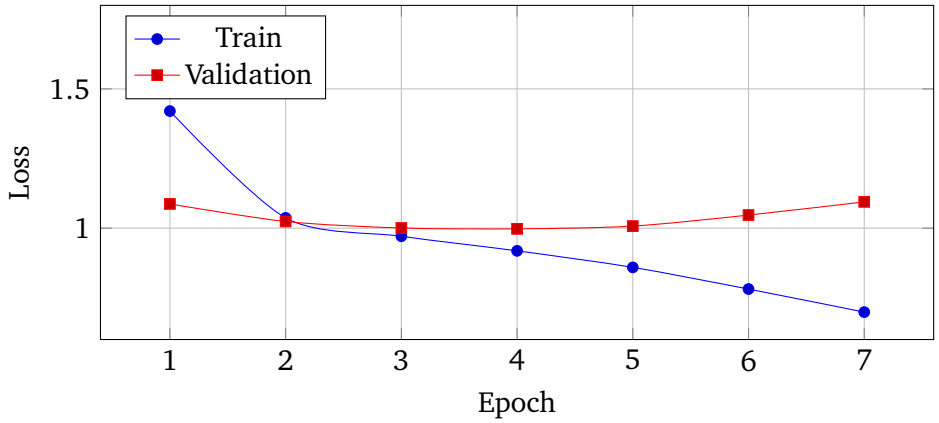


Figure 5.4: Training and validation loss per epoch for the winner trial of the SAR track.

5.5.2 Generated Captions

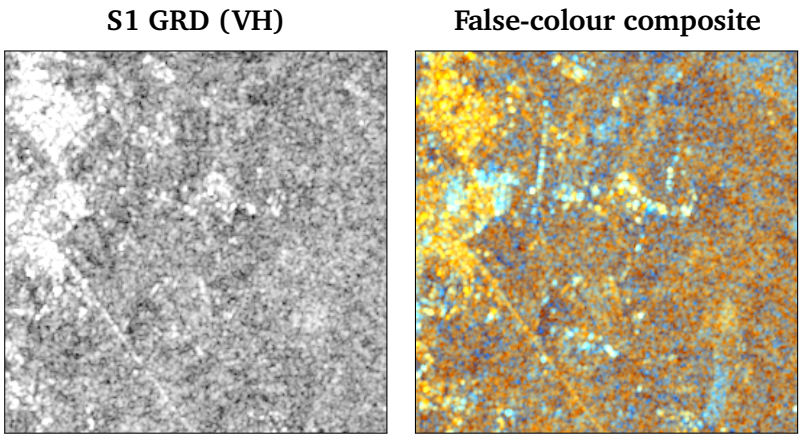
Parallel to the optical track, Figure 5.5 shows the three example images from Section 3.3 and their corresponding generated captions in the base configuration.

In the first sample, the caption describes a combination of vegetation and developed areas, and the scattering terminology is used correctly. The bright cluster in the upper left is interpreted as a double-bounce signature and associated with built-up area. The caption describes it as a localised settlement or an isolated building complex within natural surroundings, while the tile shows a town. The type of scattering signature is therefore recognised, but its spatial extent is underestimated. Unlike the optical caption for the same scene in Figure 5.3, the SAR caption does not mention roads.

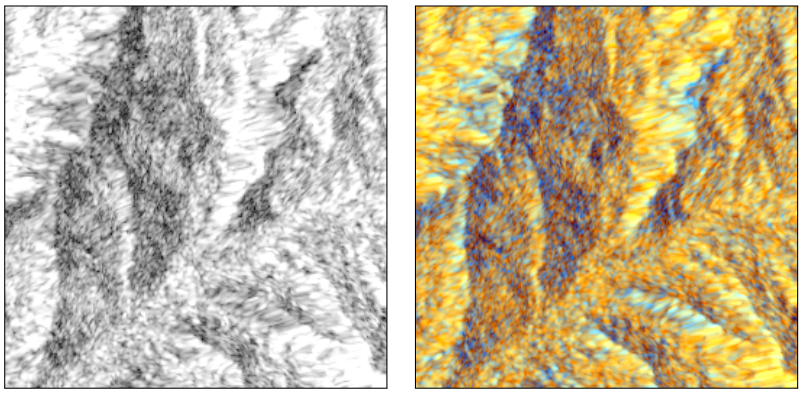
The second sample is described most accurately. The gold-coloured structures are interpreted as vegetation, while the blue areas between them are described as natural drainage features or low-lying ground with little surface roughness. Both interpretations are consistent with the image. The caption also notes that there are no concentrated double-bounce clusters and concludes that the area does not contain a settlement, which matches the scene.

The third sample is also described correctly, although the scene is not identified directly. The tile consists of a largely uniform surface with low backscatter, which the caption describes as a homogeneous area with low biomass that could represent open water or saturated, flat terrain. Open water is therefore included as a possible interpretation without being stated as the definite land-cover type. Both calm water and smooth saturated ground produce low backscatter, so the SAR measurement alone does not clearly distinguish between them (Section 3.1.3).

The optical and SAR captions of the same three scenes are compared in Section 5.7.1.

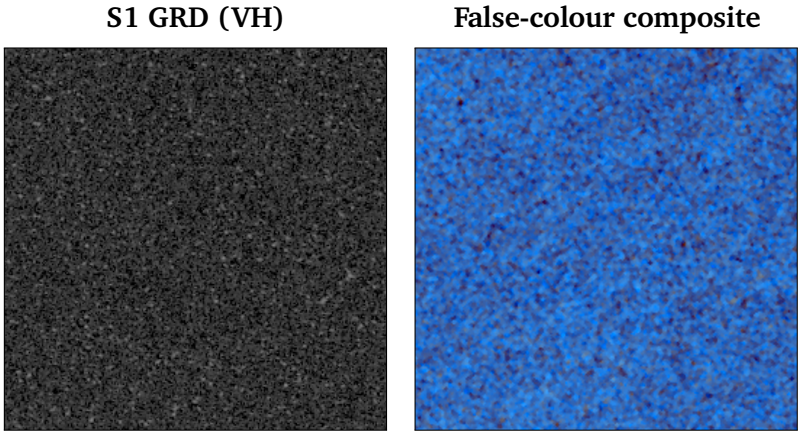


This landscape is characterized by a dense, heterogeneous mix of high-volume 3D canopy biomass and low-lying vegetation. In the upper left corner, a distinct cluster of intense white double-bounce structural reflections indicates a localized settlement or isolated building complex embedded within the natural cover. The surrounding terrain exhibits a mottled texture, transitioning from vibrant gold forest to desaturated blue areas indicative of exposed soil or sparse ground cover. These geometric signatures contrast sharply with the diffuse, grainy nature of the surrounding woodland, highlighting an area where human infrastructure has been integrated into a largely undisturbed environment.



This landscape is characterized by a high-relief mosaic of dense, 3D canopy biomass and rugged terrain. The vibrant gold textures indicate significant vegetation cover, while the interspersed blue-hued depressions represent natural drainage features or low-lying ground that lacks significant surface roughness. A careful inspection of the frame reveals no concentrated clusters of intense white double-bounce structural reflections, indicating an absence of localized settlements or engineered infrastructure within this specific area. The overall composition reflects a largely undisturbed environment shaped primarily by topographical variation rather than human land management.

Figure 5.5: Captions generated by the base configuration of the SAR track for the scenes shown in Figure 3.6. Left: VH backscatter in dB. Right: false-colour composite (R: VH, G: VV, B: VV−VH). Images adapted from [41].



The provided image represents a homogeneous expanse of low-biomass ground cover, characterized by a uniform blue texture indicative of open water or saturated, flat terrain. There is an absence of high-intensity double-bounce structural reflections, confirming the lack of engineered settlements, buildings, or infrastructure within this specific frame. Furthermore, no geometric agricultural boundaries are present in the scene, suggesting a natural landscape devoid of managed field partitioning.

Figure 5.5: Captions generated by the base configuration of the SAR track (continued).

5.5.3 Metric Results

Table 5.5: Evaluation results of the SAR track. Mean over the 5,120 test pairs with the half-width of the 95% confidence interval. The conditions are defined in Table 5.3.

	BLEU-4	ROUGE-L	METEOR	SBERT
Base (MLP)	.1181 ± .0017	.2966 ± .0019	.4066 ± .0018	.8637 ± .0018
D1 (linear)	.1177 ± .0016	.2979 ± .0018	.3976 ± .0019	.8616 ± .0019
D2 (Vicuna)	.1143 ± .0016	.2916 ± .0017	.3988 ± .0018	.8615 ± .0019
D3 (sensor-aware)	.0615 ± .0012	.2318 ± .0015	.3388 ± .0018	.7943 ± .0021
shifted-captions	.0914 ± .0014	.2577 ± .0016	.3663 ± .0018	.8286 ± .0022
Baseline	.0119 ± .0002	.1222 ± .0003	.1822 ± .0004	.5559 ± .0025

The trained conditions again consistently outperform the zero-shot baseline across all four metrics. The difference is particularly large for BLEU-4. The baseline obtains a score of 0.0119, while the base condition reaches a value that is 9.9 times higher. The corresponding factors for ROUGE-L and METEOR are 2.4 and 2.2.

For D1, the results are very similar for BLEU-4, ROUGE-L, and SBERT. Their confidence intervals overlap, with differences of only 0.0004, 0.0013, and 0.0021, respectively. ROUGE-L has a slightly higher value for the linear bridge, but the difference is not clearly resolved. With METEOR, the MLP achieves a higher score by 0.0090, and the intervals are separated. In the total setup these differences are marginal.

Replacing LLaVA-1.5 with Vicuna v1.5 lowers the scores of the base condition on BLEU-4, ROUGE-L, and METEOR by 0.0038, 0.0050, and 0.0078, respectively. The confidence intervals are separate for these three metrics, while they overlap for SBERT.

The differences are of the same order as those observed for D1 and remain far below the distance to the baseline, so neither design factor changes the generated captions to a relevant extent.

The sensor-aware prompt from Section A.2 leads to the lowest scores among the trained conditions for all four metrics. Compared with the base condition, BLEU-4 decreases by 0.0566 and the SBERT cosine by 0.0694. The BLEU-4 decrease corresponds to more than half of the 0.1063 that separates the base condition from the zero-shot baseline, so this is the only design factor whose effect is of a comparable order to the training itself, and it is roughly six times larger than the differences observed for D1 and D2. The D3 results are also below those of the shifted condition for every metric, with separate confidence intervals throughout. Even with the correct pairing, the generated captions therefore show less agreement with the reference captions under the sensor-aware instruction than under the default instruction. A look at the results reveals that a different answering pattern was used and the sentences are enumerated, as visible in the following example for the third of the example images.

```
"1.  The landscape is dominated by a uniform, low-biomass
surface characterized by specular and diffuse scattering
returns across the entire frame.  This consistent
texture indicates an open area with minimal vegetation
or complex topography, presenting as a smooth blue
expanse devoid of significant 3D canopy biomass.   2.
A meticulous scan reveals no localized settlements
or engineered structures within this scene; all corners
and edges remain free from double-bounce reflections
indicative of human development.  3.  The absence of
bright white pinpoints in the margins confirms that
there are no concentrated structural features such
as buildings or infrastructure clusters to be found
here.   4.  The overall lack of geometric boundaries
suggests a non-agricultural environment where natural
processes have shaped the terrain without interference
from managed land use. 5.  No linear low-return paths
were detected, indicating limited waterways."
```

The shifted condition scores lower than those of the base condition for every metric, with differences between 0.0268 for BLEU-4 and 0.0403 for METEOR.

5.6 Contribution of the visual input to performance

The results in Section 5.4 and Section 5.5 show how well the generated captions match the reference captions. They do not show how much of this performance is retrieved from the visual input. The reference captions all describe Sentinel scenes and therefore use a relatively limited vocabulary. In addition, the captions within each track were generated by the same process and follow a similar style, as described in Section 5.3.2. The bridge can therefore improve the decoder's predictions for this vocabulary without necessarily making the output dependent on the individual image. This section uses two additional checks to estimate how much of the measured performance can be attributed to the correct visual input.

5.6.1 Performance without the Correct Scene

The first check compares the base condition with the shifted condition introduced in section 5.2.2. The trained bridge, decoder, prompt, and decoding settings are the same for both conditions, and the generated captions are evaluated against the same reference captions. The difference lies in the pairing between the generated captions and the reference captions. In the base condition, each generated caption is compared with the reference from the same scene, while the shifted condition uses a reference from a different scene. Table 5.6 shows the scores for both conditions and the resulting difference for the optical and SAR tracks.

Table 5.6: Scores with the correct reference (base condition) and with a reference from a different scene (shifted condition), and their difference Δ. The two tracks are evaluated against different reference datasets, so the columns are read within a track and not across the two.

	Optical			SAR		
	Shifted	Base	Δ	Shifted	Base	Δ
BLEU-4	0.1274	0.1507	0.0233	0.0914	0.1181	0.0267
ROUGE-L	0.3295	0.3631	0.0337	0.2577	0.2966	0.0389
METEOR	0.2873	0.3210	0.0336	0.3663	0.4066	0.0403
SBERT	0.7100	0.7818	0.0718	0.8286	0.8637	0.0351

The base condition scores higher than the shifted condition in all eight comparisons, with separate confidence intervals throughout (table 5.4, table 5.5). The generated captions are therefore affected by the visual input, consistently across both tracks and all four metrics, and the effect exceeds any difference produced by the bridge or decoder variations.

It remains small, however, next to the distance to the zero-shot baseline. In the optical track the shifted condition already reaches a BLEU-4 of 0.1274 against 0.0260 for the baseline, and the correct reference adds only 0.0233. The SAR track behaves the same

way, with 0.0914 against 0.0119 and a further 0.0267. Most of the advantage of the trained conditions over the baseline is therefore already present when the captions are scored against a reference from another scene.

D3 is relevant here as well. It uses the correct reference yet falls below the shifted condition on all four metrics, with separate confidence intervals. Changing the instruction therefore costs more agreement with the references than withholding the correct scene. The retrieval measures in the next section examine the same conditions without relying on a single reference score.

5.6.2 Retrieval-based Check

Instead of comparing two scores, the retrieval measures rank the reference captions by their similarity to the generated caption. This also reduces the influence of the general style of the reference dataset, since a generated caption has to be more similar to its own reference than to the other references in the gallery.

The evaluation is carried out at the two levels defined in section 5.3.3, so that the results show whether a caption identifies its location and whether it also identifies the individual acquisition. Table 5.7 reports both levels for the base condition of both tracks, the conditions with the linear bridge, the SAR-specific prompt, the shifted references, and the zero-shot baseline.

The shifted condition shows that the retrieval measure behaves as expected. Its median rank is 640.0 in the optical track and 645.0 in the SAR track, compared with a chance value of 640 at scene level. At caption level, the corresponding values are 2,554.0 and 2,586.5, compared with 2,560. The shifted captions therefore produce rankings that are close to what would be expected if the ranking were independent of the generated caption and therefore also provides an empirical reference for random performance.

Both trained base conditions perform clearly better than this reference. At scene level, the optical base condition retrieves the correct scene first in 1.17% of queries, compared with 0.08% for random retrieval, and has a median rank of 188. The SAR base condition reaches 0.82% with a median rank of 290.5. Based on the median rank at scene level, the conditions form a consistent order. The optical base condition ranks first at 188, followed by the SAR base condition at 290.5, the SAR-specific prompt at 405, and finally the two shifted conditions close to the random reference.

The absolute values remain low. At scene level, the correct scene is outside the top 50 for roughly three quarters of the optical queries and about five sixths of the SAR queries. Even for the optical track, which performs better, the correct scene has a median rank of 188 out of 1,280.

The two zero-shot baselines behave differently from each other. In the SAR track, the

Table 5.7: Retrieval of the matching reference among the references of the track. The measures are defined in section 5.3.3. Chance is the value expected when the ordering is independent of the generated caption. A lower median rank is better.

	R@1	R@5	R@10	R@50	Median rank
Scene level, 1,280 scenes					
Optical, base	0.0117	0.0506	0.0842	0.2594	188.0
Optical, linear	0.0111	0.0479	0.0787	0.2432	193.0
Optical, baseline	0.0137	0.0461	0.0813	0.2486	170.0
Optical, shifted	0.0020	0.0037	0.0074	0.0398	640.0
SAR, base	0.0082	0.0305	0.0514	0.1695	290.5
SAR, linear	0.0057	0.0256	0.0426	0.1518	308.0
SAR, SAR-prompt	0.0029	0.0129	0.0229	0.1014	405.0
SAR, baseline	0.0020	0.0092	0.0186	0.0701	561.5
SAR, shifted	0.0012	0.0061	0.0104	0.0451	645.0
Chance	0.0008	0.0039	0.0078	0.0391	640.5
Caption level, 5,120 captions					
Optical, base	0.0033	0.0145	0.0258	0.0887	1,014.0
Optical, linear	0.0031	0.0131	0.0217	0.0807	1,016.0
Optical, baseline	0.0059	0.0186	0.0320	0.0926	861.0
Optical, shifted	0.0006	0.0014	0.0027	0.0115	2,554.0
SAR, base	0.0029	0.0104	0.0180	0.0619	1,299.0
SAR, linear	0.0021	0.0086	0.0145	0.0537	1,380.5
SAR, SAR-prompt	0.0012	0.0039	0.0076	0.0303	1,742.0
SAR, baseline	0.0002	0.0027	0.0047	0.0221	2,300.5
SAR, shifted	0.0006	0.0018	0.0027	0.0104	2,586.5
Chance	0.0002	0.0010	0.0020	0.0098	2,560.5

zero-shot baseline is close to random retrieval. Its caption-level R@1 is 0.0002, which corresponds to the expected value, and its median rank of 2,300.5 is only moderately better than 2,560. The trained bridge reaches a median rank of 1,299.

In the optical track, the zero-shot baseline retrieves the correct scene more often than the trained base condition at both levels. At scene level, R@1 is 0.0137 for the baseline and 0.0117 for the trained condition. At caption level, the values are 0.0059 and 0.0033. The baseline also achieves better median ranks, with 170 compared with 188 at scene level and 861 compared with 1,014 at caption level. At caption level, the baseline is better on all five reported measures. At scene level, the trained condition performs better for R@5, R@10, and R@50, while the baseline remains ahead for R@1 and the median rank. The trained condition thus leads on all four reference-based metrics in table 5.4 while falling behind the baseline on most of the retrieval measures. This is taken up in section 5.7.2.

The variant with the linear bridge was evaluated in the same way. In the optical track, the two bridge variants perform very similarly at both levels. At scene level, the median rank is 193 for the linear bridge and 188 for the MLP, while at caption level it

is 1,016 and 1,014, respectively. The MLP also has slightly higher recall for all four values. In the SAR track, the difference is marginally larger and the MLP performs better for all measures at both levels. At scene level, the linear bridge reaches an R@1 of 0.0057 compared with 0.0082 for the MLP, with median ranks of 308 and 290.5. At caption level, the corresponding values are 0.0021 and 0.0029 for R@1, and 1,380.5 and 1,299 for the median rank.

With the sensor-specific prompt, the system performs below the base condition of the SAR track at both levels. At scene level, its R@1 is 0.0029 compared with 0.0082 for the base condition, and its median rank is 405 compared with 290.5. The lower agreement with the references reported in section 5.5.3 is therefore also visible in scene identification. In this evaluation, D3 still performs clearly better than the shifted captions, so the prompt does not remove the connection to the scene completely.

The two levels differ by roughly a factor of three to four in R@1 for the trained conditions. At the caption level, the candidate set contains four acquisitions for each location, so the correct reference competes with three other captions describing the same place, whereas at scene level these three references are combined into the same target.

5.7 Discussion of the Results

After the separate presentation of the optical and SAR tracks and the examination of their alignment with the real visual input, this section discusses the main findings.

5.7.1 Comparison of Training Conditions

The reference metrics show only small differences between the linear and MLP bridge. In cases where the confidence intervals are separated, the difference is about one order of magnitude smaller than the difference between the base condition and the zero-shot baseline. The clearest example is discussed in Section 5.4.3. The metric that separates the two bridge variants also differs between the optical and SAR tracks. Given the small size of these differences, this pattern should not be overinterpreted. The retrieval results provide a slightly different picture for the SAR track. Here, the MLP variant performs better across the retrieval measures, with the clearest differences in the scene-level recalls, while the reference metrics show a separated difference for only one metric (Section 5.6.2). This suggests that the second bridge layer may have an effect that is not fully reflected by the reference metrics. However, the effect remains small, and the retrieval results are based on point estimates. In the optical track, the comparison between the two bridge variants is largely unchanged across the evaluation.

A similar pattern can be seen for the two decoder variants. The differences are small and do not provide enough evidence to favour one decoder or to explain why one variant performs better. Overall, both decoders produce results of similar quality when the decoder is frozen and only the bridge is trained.

The SAR-specific prompt gives a different result. It achieves the lowest scores among all trained conditions and is also below the shifted condition on every metric (Section 5.5.3). Since the instruction is the only component that changes, this result shows that the metrics are sensitive to the type of caption produced by the instruction. It does not mean that the sensor-aware captions describe the scenes less accurately. Instead, they are less similar to the reference captions used for evaluation. The instruction "`Describe this scene in 3–5 sentences.`" seems to encourage the model to move away from the fixed structure found in the training captions. This can lead to more free-form descriptions, but such differences are penalised by the reference-based metrics defined in Section 5.3.1. The retrieval results in Section 5.6.2 suggest that the effect is not only related to wording, since the sensor-aware condition also retrieves the correct scene less reliably than the base condition. Providing SAR-specific information through the instruction therefore does not replace the information contained in the visual tokens. This result only applies to the instruction formulation used in this study, described in Section A.2. Other instruction formulations were not tested and could produce different results.

5.7.2 Contribution of the Bridge Training

Section 5.6.1 shows that most of the advantage of the trained bridge over the zero-shot baseline is still present when the generated captions are compared with a reference from another scene. The trained bridge has shifted the captions towards the vocabulary and style of the reference dataset, which was already identified as one source of the difference to the baseline in Section 5.2.3. However, the shifted condition should not be considered a random reference point, since a mismatched reference can still describe a similar scene (Section 5.2.2). The size of the difference between the shifted condition and the baseline therefore cannot by itself show how strongly the captions depend on the individual scene. The retrieval results in Section 5.6.2 provide an additional reference point and show that the improvement has a different meaning for Sentinel-1 and Sentinel-2.

For SAR, the zero-shot baseline is close to random retrieval, while the trained bridge performs clearly better at both levels. This suggests that the trained pipeline provides information about the individual scene that the vision-language model does not obtain from the false-colour image alone. This is also the situation that motivates the proposed approach. It is consistent with the setup described in Section 5.2.3, since the vision tower of the baseline was trained on natural images and the VH/VV

representation used in this study differs from the imagery used during its original training.

The optical results lead to a different conclusion. Here, the zero-shot baseline and the trained base condition cannot be distinguished by the retrieval measures at either scene level and the baseline even performs better when considering retrieval from all acquisitions, while the trained condition leads to much better results on all four reference metrics, with clearly separated confidence intervals. The two evaluation approaches measure different properties. Training the bridge makes the captions more similar to the vocabulary and style of the reference dataset, which improves the reference metrics without showing a measurable improvement in how well a caption identifies its own scene. The two evaluation approaches can therefore rank the same conditions differently.

There are two important restrictions to this comparison. The baseline is evaluated without training and has not been adapted to the caption dataset, so the two pathways are not evaluated under identical conditions. In addition, the retrieval measures only show how well a caption can be distinguished from the other references. They do not indicate whether the description is actually correct. The optical result therefore does not show that either pathway produces more accurate captions. It shows that the advantage of the trained conditions on the reference metrics cannot by itself be interpreted as evidence for more scene-specific descriptions.

This does not remove the motivation for using a remote-sensing encoder for optical acquisitions. The SSL4EO-S12 captions were generated from the RGB channels of the Sentinel-2 acquisitions alone (Section 3.3.1) and therefore do not describe information that comes from the remaining bands. The supervision is therefore well matched to what the natural-image vision tower of the baseline can already represent, while the twelve-band input available to the geospatial encoder is not required by the training target. In this sense, the optical comparison is performed on the baseline's own terms. The study still provides a proof of concept that captions can be generated from the embeddings produced by the TerraMind encoder on optical scenes.

Taken together, the experiments give little indication that the transfer into the language space is the main limitation of the system. A small trainable bridge is sufficient to generate remote-sensing descriptions from a frozen encoder and a frozen language model for both Sentinel-1 and Sentinel-2. Neither the additional bridge capacity nor the choice between the two decoders leads to a substantial change in the results. The remaining limitations may therefore lie earlier in the pipeline, potentially in the encoder representation or in the supervision provided by the caption datasets. Since the encoder was not varied in the experiments, this cannot be separated further here.

5.7.3 Strengths and Weaknesses of the Generated Captions

Overall, the generated captions are fluent and usually identify the main type of scene correctly. For SAR, the scattering terminology is mostly consistent with the observed composite. When the measurement does not allow a clear distinction, the captions sometimes include alternative interpretations, as in the third example of Figure 5.5. The statements about the absence of built-up areas are also correct in the examples where they apply. In the optical track, the main land-cover categories are generally identified correctly, and individual objects can also be mentioned when they are clearly visible against a relatively homogeneous background.

The descriptions are nevertheless quite coarse, and important structures are sometimes left out. This can be seen in the first optical example, where the stadium and additional roads are not mentioned. A possible reason is the token structure described in Section 4.2.2. Each visual token covers an area of 160×160 metres, meaning that a stadium can be represented together with other land-cover information in the same token. A small feature may therefore have less influence on the generated caption when most of the token is covered by another land-cover type. The third optical example gives a useful contrast. Here, a few bright boats are clearly visible against a homogeneous dark surface, allowing the corresponding visual token to carry a strong signal. Both boats and the colour of one of them are included in the caption. More specific descriptions therefore seem possible when individual objects are clearly separated from their surroundings, although the examples do not show whether this can also be achieved in more complex scenes.

Another limitation is that parts of the captions contain information that is not very specific to the individual scene. Some statements are too general to be verified reliably from the image, while descriptions of absent features provide little information because they occur in many other scenes. As a result, the captions rarely contain enough details to clearly distinguish one scene from another with a similar overall appearance.

There are also cases where features are described that cannot actually be seen in the image. In the second optical example, for instance, the caption mentions two waterways. This interpretation is understandable because the brown areas form a branching structure that resembles a drainage network, but no waterways are actually visible. Other examples that are not shown here contain the names of parks, which cannot be identified from satellite imagery alone. The caption dataset provides a possible explanation for this behaviour because additional information from Overture Maps is used when creating the captions (Section 3.3.1). Place names can therefore become part of the training supervision even when they are not supported by the image itself.

The training data also influence the way the captions are formulated. Instead of producing completely free-form descriptions, the model often follows the phrasing

patterns found in the reference captions. This is in contrast to the intention described in Section 2.6, where free-form language generation was one of the three properties that the approach was intended to combine. A likely reason can be seen in the reference captions themselves. Many of them begin with similar sequences of words, as shown in Figure 3.5 and Figure 3.6. Optical captions frequently use the pattern "`The image is a satellite view of ...`", while SAR captions often begin with "`The landscape is characterized by a ...`". Some of this regularity is probably also caused by the way the captions were generated. The prompt for the SAR captions (Section A.1) specifies how the caption should start, provides a fixed set of physical descriptors, and gives an example for the expected writing style. The Llama3-SSL4EO-S12 captions were also generated using a prompted model (Section 3.3.1). The similar phrasing can therefore not only be explained by similarities between the scenes, but also by the prompts that were used to generate the descriptions. Since the model is trained with a next-token objective, these recurring patterns are reinforced during training. Moving away from them can increase the loss, which gives the model a reason to reproduce the structure. The result from the SAR-specific prompt is consistent with this explanation. The instruction used there encourages a less template-based description, but this condition receives lower scores on all reference metrics (Section 5.7.1).

6 Conclusion

6.1 Summary

The combination of properties identified as missing in the formulated Research Gap from Section 2.6 was implemented in this work. A frozen TerraMind encoder and a frozen LLaVA-1.5 decoder, connected by a small trainable projection, can generate fluent descriptions of Sentinel scenes and correctly identify their main landscape type. For Sentinel-1, this is possible without optical input at any stage, since the supervision was generated directly from the radar composite. The experiments therefore show that the alignment stage can transfer information from a geospatial encoder into a language model, at least to the extent that usable scene descriptions can be generated.

The two additional evaluations put these results into context. The reference metrics alone give a more positive impression of the trained system than the complete evaluation supports. Generated captions consistently depend on the scene they were produced from in both tracks, but this effect is small compared with the difference between the trained system and the zero-shot baseline. A substantial part of this difference comes from the projection adapting the captions to the vocabulary and sentence structures of the reference dataset. As a result, a caption can achieve a higher reference score because it follows the style of the dataset without necessarily describing the individual scene in more detail. The reference-based metrics cannot distinguish between these cases. The retrieval evaluation provides an additional measure with a defined chance level and makes this distinction more visible.

The difference between the two tracks is particularly clear for Sentinel-1. The zero-shot vision-language model performs close to chance, while the trained bridge achieves noticeably better retrieval results. This corresponds to the main motivation for the proposed approach, since SAR imagery is different from the natural images used to pretrain the vision component of the baseline. Sentinel-2 shows a different result. The trained and zero-shot pathways identify the correct scene with similar performance in the retrieval evaluation. This does not mean that the geospatial encoder provides no benefit for optical imagery. The optical supervision was generated from the RGB channels alone and therefore does not require information from the remaining Sentinel-2 bands.

The results also give an indication of where the main limitation of the current system may be. Changing the capacity of the projection or using a different frozen decoder has only a small effect on the results. The transfer into the language space, which was considered a possible source of information loss in Section 1.1, therefore does

not appear to be the main limiting factor in the tested setup. The caption supervision seems to be more relevant. It only describes three of the twelve Sentinel-2 bands, and its regular phrasing encourages the model to reproduce common sentence structures. This can be seen in the repeated opening phrases in the generated captions and in the lower reference scores obtained when the instruction encourages more free-form descriptions. These limitations are related to the supervision and could be addressed without changing the frozen encoder or decoder.

The conclusions are limited by the experimental setup. Each condition was trained only once, so the experiments do not provide an estimate of run-to-run variation for the small differences between conditions. In addition, each acquisition is evaluated against a single reference caption produced by one caption generation process. This affects the absolute metric values and favours the style that the model learns from the training data. The results therefore provide an indication of where the main limitation of the system may lie, but they do not support a reliable ranking of the individual design variants.

6.2 Future Work

The design of this work keeps several factors fixed in order to isolate the effect of the alignment step. The encoder and language model remain frozen, Sentinel-1 and Sentinel-2 are evaluated separately, and the training data provide one caption per acquisition. These choices define the scope of the current study and also point to several possible extensions. The following directions build on the current pipeline and propose more points of variation to try and further improve the results.

6.2.1 Broader and task-oriented datasets

The experiments in Chapter 5 use a single dataset. SSL4EO-S12 provides global coverage and is not restricted to a particular region, but it contains generic scene descriptions and the two caption sets described in Section 3.3 provide only one reference per acquisition. This limits the interpretation of caption metrics. With a single reference, a correct caption can receive a low score simply because it uses different wording.

Future work could address both limitations by using datasets with multiple references or with captions generated independently of the current captioning pipeline. ChatEarthNet [100] provides an independent optical caption dataset for Sentinel-2, while SAR-TEXT [36] and SARChat-2M [101] provide language data associated with SAR imagery. On the task side, datasets with structured labels, such as the flood datasets used in operational applications [102, 103], would allow generated descrip-

tions to be evaluated against spatial ground truth instead of only against another sentence.

A further limitation is the overlap between the data used in this thesis and TerraMind's pretraining data. As discussed in Section 3.2, TerraMesh contains data from some of the same Sentinel grids as SSL4EO-S12. Keeping the original SSL4EO-S12 splits unchanged ensures that the validation scenes do not come from the training split used in this thesis, but it does not completely exclude their presence in TerraMind's pretraining data. A dataset that is disjoint from TerraMesh would remove this potential source of contamination.

6.2.2 Instruction tuning and parameter-efficient adaptation of the decoder

The current setup excludes LoRA [16] because the aim is to isolate the contribution of the bridge. Introducing decoder adaptation would add further hyperparameters and make it more difficult to attribute improvements to the visual-language alignment itself. Once the research question shifts from the capabilities of a bridge with a frozen language model to the maximum captioning performance that can be achieved, adapting the language model becomes a natural next step.

A possible extension would follow the two-stage training strategy used by LLaVA [21, 23]. The bridge would first be trained as in the present work, followed by a second stage in which a parameter-efficient adaptation method such as LoRA is applied to the language model. The bridge and decoder could then be trained jointly on instruction-formatted data, similar to the approach used by RS-LLaVA for optical remote sensing [47]. This extension would require additional supervision because the current caption dataset only teaches the model to describe an image. An instruction dataset would allow the model to answer questions and perform different tasks based on the same visual input.

Decoder adaptation would also affect the interpretation of the grounding analysis introduced in Section 4.5. With a frozen decoder, changes in the loss can be related directly to the contribution of the visual tokens under a fixed language-model prior. Fine-tuning the decoder changes this prior as well. A future experiment should therefore evaluate the grounding gap against a decoder that was adapted under the same conditions. Caption quality alone would not show whether the additional capacity improves the use of visual information or mainly improves the language model's ability to generate fluent text.

6.2.3 A language-aligned SAR encoder as a point of comparison

TerraMind is used in this thesis because its encoder was pretrained jointly on multiple remote-sensing modalities, including Sentinel-1 and Sentinel-2, while the language component was also included in the pretraining process. This provides a remote-sensing-specific representation that can subsequently be connected to a language model through the bridge. Contrastive models provide a useful comparison because their training explicitly aligns visual and textual representations.

A future experiment could keep the bridge, language model, captions, data splits, prompts, and optimisation procedure fixed and replace only the visual encoder. Llama3-MS-CLIP [52] provides a suitable candidate for the optical branch because it is contrastively trained on Sentinel-2 data using LLM-generated captions. A corresponding SAR experiment could use a SAR-language encoder such as SARCLIP [33] or an encoder trained using SAR-TEXT [36]. The two modalities would remain separate, consistent with the experimental design of this thesis.

Such a comparison requires careful consideration of data overlap. Llama3-MS-CLIP is trained using SSL4EO-S12 and generated captions, so some scenes from the evaluation data may also occur in its pretraining data. Its results would therefore have to be interpreted with this overlap in mind. A dataset that is independent of both TerraMesh and SSL4EO-S12 would provide a cleaner basis for the comparison. The candidate encoders also differ in embedding dimension and token count. The bridge could therefore be kept identical in terms of depth, activation, and regularisation while adapting its input dimension to the respective encoder.

6.2.4 Joint Sentinel-1 and Sentinel-2 input

The two sensor streams are kept separate in the current experiments. This choice follows from the available caption supervision. The captions describe information visible in the respective input image, so a model receiving both Sentinel-1 and Sentinel-2 could rely mainly on the optical input. Any improvement from feature fusion would consequently be difficult to attribute to the contribution of SAR.

A future experiment could address this issue by creating three types of captions for the same acquisitions. One caption set would describe the Sentinel-1 observation, another would describe the Sentinel-2 observation, and a third would describe information that becomes available only when both observations are considered together. Examples include flooding beneath cloud cover, changes in scattering behaviour that are not apparent in optical imagery, or structures whose spatial extent is visible in one modality while their surface characteristics are more apparent in the other. The training objective could then select the corresponding reference caption according to the modalities available to the model.

This setup would make it possible to test whether fusion provides information that neither modality can provide independently. It would also allow the model to operate with Sentinel-1 alone, Sentinel-2 alone, or both modalities together. A modality mask could be sampled during training so that the model encounters all three configurations regularly.

The implementation could build on the existing encoder wrapper and training procedure. TerraMind already supports different combinations of modalities through its modality sampling strategy [38]. The main additional requirement would be the creation of captions that explicitly depend on the available modality combination. Two architectural variants would be particularly relevant. One could use a single bridge for the concatenated token sequence, while the other could assign separate bridges to the two modalities before mapping them into the shared language space.

6.2.5 Using TerraMind's generative capabilities

This work uses TerraMind only as a source of visual representations. Its generative capabilities could provide another way to connect SAR observations with information that is easier for a language model to interpret. TerraMind can generate one modality from another, which makes it possible to use an additional modality as an intermediate representation [38].

For Sentinel-1, this could involve generating an intermediate Sentinel-2 or land-cover representation and providing it together with the original SAR features to the language model. Such an intermediate representation could make information that is difficult to express directly from SAR more accessible to the bridge. It could also provide a way to test whether the limitation of SAR captioning originates from the SAR representation itself or from the mapping between SAR features and language.

The quality of the generated intermediate representation would have to be evaluated separately. Errors in the generated modality could otherwise be transferred to the final caption and become difficult to distinguish from errors introduced during language alignment. Future experiments should therefore compare generated intermediate representations with the corresponding reference data and include an upper-bound experiment using the true modality. The additional generation step would also increase computational cost, particularly if it were performed during training. Generating and storing the intermediate representations before bridge training would be one possible solution.

These directions extend different parts of the current pipeline. New datasets would improve the supervision and evaluation of the language alignment. Decoder adaptation would test how much additional performance can be obtained once the language model is allowed to change. Encoder comparisons would show how the choice

of visual representation affects the bridge. Joint Sentinel-1 and Sentinel-2 training would investigate whether the two modalities can provide complementary information when the language supervision explicitly requires it. Finally, TerraMind's generative capabilities could be used to study whether intermediate representations can help transfer information from SAR into the language model. Together, these extensions would move beyond the controlled bridge-only setting of this thesis and examine how the individual components contribute to a more capable geospatial vision-language system.

Bibliography

[1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep Learning”. In: *Nature* 521.7553 (May 2015), pp. 436–444. ISSN: 1476-4687. DOI: 10.1038/nature14539. (Visited on 08/05/2026).

[2] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. Adaptive Computation and Machine Learning. Cambridge, Massachusetts: The MIT Press, 2016. ISBN: 978-0-262-03561-3.

[3] Ashish Vaswani et al. “Attention Is All You Need”. In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc., 2017. (Visited on 08/01/2026).

[4] Dan Hendrycks and Kevin Gimpel. *Gaussian Error Linear Units (GELUs)*. June 2023. DOI: 10.48550/arXiv.1606.08415. arXiv: 1606.08415 [cs.LG]. (Visited on 08/01/2026).

[5] Alexey Dosovitskiy et al. “An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: *International Conference on Learning Representations*. Oct. 2021. (Visited on 08/01/2026).

[6] Rishi Bommasani et al. *On the Opportunities and Risks of Foundation Models*. July 2022. DOI: 10.48550/arXiv.2108.07258. arXiv: 2108.07258 [cs.LG]. (Visited on 08/01/2026).

[7] Aoran Xiao et al. “Foundation Models for Remote Sensing and Earth Observation: A Survey”. In: (June 2025). arXiv: 2410.16602.

[8] Ting Chen et al. “A Simple Framework for Contrastive Learning of Visual Representations”. In: *Proceedings of the 37th International Conference on Machine Learning*. PMLR, Nov. 2020, pp. 1597–1607. (Visited on 08/08/2026).

[9] Kaiming He et al. “Momentum Contrast for Unsupervised Visual Representation Learning”. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020, pp. 9726–9735. DOI: 10.1109/CVPR42600.2020.00975. (Visited on 08/08/2026).

[10] Alec Radford et al. “Learning Transferable Visual Models From Natural Language Supervision”. In: *Proceedings of the 38th International Conference on Machine Learning*. PMLR, July 2021, pp. 8748–8763. (Visited on 07/22/2026).

[11] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. May 2019. DOI: 10.48550/arXiv.1810.04805. arXiv: 1810.04805 [cs.CL]. (Visited on 08/08/2026).

[12] Kaiming He et al. “Masked Autoencoders Are Scalable Vision Learners”. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2022, pp. 15979–15988. DOI: 10.1109/CVPR52688.2022.01553. (Visited on 07/06/2026).

[13] Yezhen Cong et al. “SatMAE: Pre-Training Transformers for Temporal and Multi-Spectral Satellite Imagery”. In: *Advances in Neural Information Processing Systems 35*. New Orleans, Louisiana, USA: Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2022, pp. 197–211. ISBN: 978-1-7138-7108-8. DOI: 10.52202/068431-0015. (Visited on 08/08/2026).

[14] Danfeng Hong et al. “SpectralGPT: Spectral Remote Sensing Foundation Model”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46.8 (Aug. 2024), pp. 5227–5244. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2024.3362475. (Visited on 08/08/2026).

[15] Alec Radford et al. “Improving Language Understanding by Generative Pre-Training”. In: (2018).

[16] Edward J. Hu et al. “LoRA: Low-Rank Adaptation of Large Language Models”. In: *International Conference on Learning Representations*. Jan. 2022. (Visited on 08/17/2026).

[17] Tom Brown et al. “Language Models Are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems*. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. (Visited on 08/08/2026).

[18] Hugo Touvron et al. *LLaMA: Open and Efficient Foundation Language Models*. 2023. DOI: 10.48550/ARXIV.2302.13971. (Visited on 08/01/2026).

[19] Hugo Touvron et al. *Llama 2: Open Foundation and Fine-Tuned Chat Models*. July 2023. DOI: 10.48550/arXiv.2307.09288. arXiv: 2307.09288 [cs.CL]. (Visited on 08/01/2026).

[20] Wei-Lin Chiang et al. *Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality*. https://www.lmsys.org/blog/2023-03-30-vicuna/. Mar. 2023. (Visited on 08/08/2026).

[21] Haotian Liu et al. “Improved Baselines with Visual Instruction Tuning”. In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2024, pp. 26286–26296. DOI: 10.1109/CVPR52733.2024.02484. (Visited on 06/20/2026).

[22] Yan Shu et al. *EarthMind: Leveraging Cross-Sensor Data for Advanced Earth Observation Interpretation with a Unified Multimodal LLM*. Sept. 2025. DOI: 10.48550/arXiv.2506.01667. arXiv: 2506.01667 [cs.CV]. (Visited on 06/10/2026).

[23] Haotian Liu et al. “Visual Instruction Tuning”. In: *Advances in Neural Information Processing Systems* 36 (Dec. 2023), pp. 34892–34916. (Visited on 06/13/2026).

[24] Junnan Li et al. “BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models”. In: *Proceedings of the 40th International Conference on Machine Learning*. PMLR, July 2023, pp. 19730–19742. (Visited on 07/22/2026).

[25] Xin Guo et al. “SkySense: A Multi-Modal Remote Sensing Foundation Model Towards Universal Interpretation for Earth Observation Imagery”. In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2024, pp. 27662–27673. DOI: 10.1109/CVPR52733.2024.02613. (Visited on 08/05/2026).

[26] Yi Wang et al. “Towards a Unified Copernicus Foundation Model for Earth Vision”. In: *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2025, pp. 9888–9899. DOI: 10.1109/ICCV51701.2025.00922. (Visited on 08/11/2026).

[27] Fan Liu et al. “RemoteCLIP: A Vision Language Foundation Model for Remote Sensing”. In: *IEEE Transactions on Geoscience and Remote Sensing* 62 (2024), pp. 1–16. ISSN: 1558-0644. DOI: 10.1109/TGRS.2024.3390838. (Visited on 08/05/2026).

[28] Xiang Li et al. “RS-CLIP: Zero Shot Remote Sensing Scene Classification via Contrastive Vision-Language Supervision”. In: *International Journal of Applied Earth Observation and Geoinformation* 124 (Nov. 2023), p. 103497. ISSN: 15698432. DOI: 10.1016/j.jag.2023.103497. (Visited on 08/05/2026).

[29] Zilun Zhang et al. “RS5M and GeoRSCLIP: A Large-Scale Vision- Language Dataset and a Large Vision-Language Model for Remote Sensing”. In: *IEEE Transactions on Geoscience and Remote Sensing* 62 (2024), pp. 1–23. ISSN: 1558-0644. DOI: 10.1109/TGRS.2024.3449154. (Visited on 08/02/2026).

[30] Zhitong Xiong et al. *DOFA-CLIP: Multimodal Vision-Language Foundation Models for Earth Observation*. July 2025. DOI: 10.48550/arXiv.2503.06312. arXiv: 2503.06312 [cs.CV]. (Visited on 09/03/2026).

[31] Angelos Zavras et al. “Mind the Modality Gap: Towards a Remote Sensing Vision-Language Model via Cross-Modal Alignment”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 228 (Oct. 2025), pp. 270–287. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2025.06.019. (Visited on 08/11/2026).

[32] Minh Kha Do et al. *Spectrally Distilled Representations Aligned with Instruction-Augmented LLMs for Satellite Imagery*. Feb. 2026. DOI: 10.48550/arXiv.2602.22613. arXiv: 2602.22613 [cs.CV]. (Visited on 08/05/2026).

[33] Pengfei Wang et al. “SARCLIP: The First Vision Language Foundation Model for SAR Image”. In: *IEEE Transactions on Geoscience and Remote Sensing* (2025). ISSN: 15580644. DOI: 10.1109/TGRS.2025.3630131.

[34] Chaowei Jiang et al. “SARCLIP: A Multimodal Foundation Framework for SAR Imagery via Contrastive Language-Image Pre-Training”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 231 (Jan. 2026), pp. 17–34. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2025.10.017. (Visited on 08/05/2026).

[35] Qiwei Ma et al. *SARVLM: A Vision Language Foundation Model for Semantic Understanding in SAR Imagery*. May 2026. DOI: 10.48550/arXiv.2510.22665. arXiv: 2510.22665 [cs.CV]. (Visited on 08/05/2026).

[36] Yiguo He et al. *SAR-TEXT: A Large-Scale SAR Image-Text Dataset Built with SAR-Narrator and A Progressive Learning Strategy for Downstream Tasks*. Oct. 2025. DOI: 10.48550/arXiv.2507.18743. arXiv: 2507.18743 [cs.CV]. (Visited on 08/05/2026).

[37] Yimin Wei et al. “SARLANG-1M: A Benchmark for Vision–Language Modeling in SAR Image Understanding”. In: *IEEE Transactions on Geoscience and Remote Sensing* 64 (2026), pp. 1–20. ISSN: 1558-0644. DOI: 10.1109/TGRS.2026.3652099. (Visited on 08/11/2026).

[38] Johannes Jakubik et al. “TerraMind: Large-Scale Generative Multimodality for Earth Observation”. In: *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2025, pp. 7383–7394. DOI: 10.1109/ICCV51701.2025.00693. (Visited on 06/10/2026).

[39] Valerio Marsocci et al. *PANGAEA: A Global and Inclusive Benchmark for Geospatial Foundation Models*. Apr. 2025. DOI: 10.48550/arXiv.2412.04204. arXiv: 2412.04204 [cs.CV]. (Visited on 09/30/2026).

[40] Johannes Jakubik et al. *TerraMind: Large-Scale Generative Multimodality for Earth Observation (Supplementary Material)*. June 2026. DOI: 10.48550/arXiv.2504.11171. arXiv: 2504.11171 [cs.CV]. (Visited on 07/22/2026).

[41] Benedikt Blumenstiel et al. *SSL4EO-S12 v1.1: A Multimodal, Multiseasonal Dataset for Pretraining, Updated*. Feb. 2026. DOI: 10.48550/arXiv.2503.00168. arXiv: 2503.00168 [cs.CV]. (Visited on 07/16/2026).

[42] Alistair Francis and Mikolaj Czerkawski. “Major TOM: Expandable Datasets for Earth Observation”. In: *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*. July 2024, pp. 2935–2940. DOI: 10.1109/IGARSS53475.2024.10640760. (Visited on 08/18/2026).

[43] *Terratorch/Terratorch/Models/Backbones/Terramind/Model at Main · Torchgeo/Terratorch*. https://github.com/torchgeo/terratorch/tree/main/terratorch/models/backbones/terramind/model. (Visited on 08/11/2026).

[44] IBM and ESA. *TerraMind 1.0 GitHub Repository*. International Business Machines. July 2026. (Visited on 07/22/2026).

[45] Yuan Hu et al. "RSGPT: A Remote Sensing Vision Language Model and Benchmark". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 224 (June 2025), pp. 272–286. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2025.03.028. (Visited on 08/05/2026).

[46] Kartik Kuckreja et al. "GeoChat:Grounded Large Vision-Language Model for Remote Sensing". In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2024, pp. 27831–27840. DOI: 10.1109/CVPR52733.2024.02629. (Visited on 08/05/2026).

[47] Yakoub Bazi et al. "RS-LLaVA: A Large Vision-Language Model for Joint Captioning and Question Answering in Remote Sensing Imagery". In: *Remote Sensing* 16.9 (Jan. 2024), p. 1477. ISSN: 2072-4292. DOI: 10.3390/rs16091477. (Visited on 08/05/2026).

[48] Yang Zhan, Zhitong Xiong, and Yuan Yuan. "SkyEyeGPT: Unifying Remote Sensing Vision-Language Tasks via Instruction Tuning with Large Language Model". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 221 (Mar. 2025), pp. 64–77. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2025.01.020. (Visited on 08/05/2026).

[49] Zhenshi Li et al. "LHRS-Bot-Nova: Improved Multimodal Large Language Model for Remote Sensing Vision-Language Interpretation". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 227 (Sept. 2025), pp. 539–550. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2025.06.003. (Visited on 08/05/2026).

[50] Peijin Wang et al. "RingMoGPT: A Unified Remote Sensing Foundation Model for Vision, Language, and Grounded Tasks". In: *IEEE Transactions on Geoscience and Remote Sensing* 63 (2025), pp. 1–20. ISSN: 1558-0644. DOI: 10.1109/TGRS.2024.3510833. (Visited on 08/05/2026).

[51] Junwei Luo et al. *SkySenseGPT: A Fine-Grained Instruction Tuning Dataset and Model for Remote Sensing Vision-Language Understanding*. July 2024. DOI: 10.48550/arXiv.2406.10100. arXiv: 2406.10100 [cs.CV]. (Visited on 08/05/2026).

[52] Clive Tinashe Marimo et al. "Beyond the Visible: Multispectral Vision-Language Learning for Earth Observation". In: *Machine Learning and Knowledge Discovery in Databases. Research Track*. Ed. by Rita P. Ribeiro et al. Vol. 16018. Cham: Springer Nature Switzerland, 2026, pp. 359–375. ISBN: 978-3-032-06105-8. DOI: 10.1007/978-3-032-06106-5_21. (Visited on 06/22/2026).

[53] Wei Zhang et al. "EarthGPT: A Universal Multimodal Large Language Model for Multisensor Image Comprehension in Remote Sensing Domain". In: *IEEE Transactions on Geoscience and Remote Sensing* 62 (2024), pp. 1–20. ISSN: 1558-0644. DOI: 10.1109/TGRS.2024.3409624. (Visited on 08/05/2026).

[54] Wei Zhang et al. “EarthGPT-X: A Spatial MLLM for Multilevel Multisource Remote Sensing Imagery Understanding With Visual Prompting”. In: *IEEE Transactions on Geoscience and Remote Sensing* 63 (2025), pp. 1–21. ISSN: 1558-0644. DOI: 10.1109/TGRS.2025.3626941. (Visited on 08/05/2026).

[55] Sagar Soni et al. “EarthDial: Turning Multi-sensory Earth Observations to Interactive Dialogues”. In: *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2025, pp. 14303–14313. DOI: 10.1109/CVPR52734.2025.01334. (Visited on 08/05/2026).

[56] Thomas M. Lillesand, Ralph W. Kiefer, and Jonathan W. Chipman. *Remote Sensing and Image Interpretation*. Seventh edition. Hoboken, N.J: John Wiley & Sons, Inc, 2015. ISBN: 978-1-118-34328-9.

[57] James B. Campbell and Randolph H. Wynne. *Introduction to Remote Sensing*. Fifth edition. New York London: The Guilford Press, 2011. ISBN: 978-1-60918-176-5.

[58] Curtis E. Woodcock et al. “Free Access to Landsat Imagery”. In: *Science* 320.5879 (May 2008), pp. 1011–1011. DOI: 10.1126/science.320.5879.1011a. (Visited on 07/24/2026).

[59] Publications Office of the European Union. *European Earth Monitoring Programme (GMES) | EUR-Lex*. https://eur-lex.europa.eu/EN/legal-content/summary/european-earth-monitoring-programme-gmes.html. (Visited on 07/24/2026).

[60] European Parliament and Council of the European Union. *Regulation (EU) 2021/696 of the European Parliament and of the Council of 28 April 2021 Establishing the Union Space Programme and the European Union Agency for the Space Programme and Repealing Regulations (EU) No 912/2010, (EU) No 1285/2013 and (EU) No 377/2014 and Decision No 541/2014/EU*. Apr. 2021. (Visited on 07/24/2026).

[61] Copernicus Data Space Ecosystem. *Copernicus Data Space Ecosystem*. https://dataspace.copernicus.eu/about. (Visited on 07/25/2026).

[62] M. Drusch et al. “Sentinel-2: ESA’s Optical High-Resolution Mission for GMES Operational Services”. In: *Remote Sensing of Environment* 120 (May 2012), pp. 25–36. ISSN: 00344257. DOI: 10.1016/j.rse.2011.11.026. (Visited on 07/28/2026).

[63] Charles Elachi and Jakob Van Zyl. *Introduction to the Physics and Techniques of Remote Sensing*. 3rd ed. Wiley Series in Remote Sensing and Image Processing Ser. Hoboken: Wiley, 2021. ISBN: 978-1-119-52301-7. DOI: 10.1002/9781119523048.

[64] European Space Agency. *SentiWiki*. https://sentiwiki.copernicus.eu/web/sentiwiki. (Visited on 07/24/2026).

[65] Susan K. Meerdink et al. “The ECOSTRESS Spectral Library Version 1.0”. In: *Remote Sensing of Environment* 230 (Sept. 2019), p. 111196. ISSN: 00344257. DOI: 10.1016/j.rse.2019.05.015. (Visited on 07/28/2026).

[66] A.M. Baldridge et al. “The ASTER Spectral Library Version 2.0”. In: *Remote Sensing of Environment* 113.4 (Apr. 2009), pp. 711–715. ISSN: 00344257. DOI: 10.1016/j.rse.2008.11.007. (Visited on 07/28/2026).

[67] Ramon Torres et al. “GMES Sentinel-1 Mission”. In: *Remote Sensing of Environment* 120 (May 2012), pp. 9–24. ISSN: 00344257. DOI: 10.1016/j.rse.2011.05.028. (Visited on 07/25/2026).

[68] Alberto Moreira et al. “A Tutorial on Synthetic Aperture Radar”. In: *IEEE Geoscience and Remote Sensing Magazine* 1.1 (Mar. 2013), pp. 6–43. ISSN: 2168-6831. DOI: 10.1109/MGRS.2013.2248301. (Visited on 07/25/2026).

[69] C. Pohl and J. L. Van Genderen. “Review Article Multisensor Image Fusion in Remote Sensing: Concepts, Methods and Applications”. In: *International Journal of Remote Sensing* 19.5 (Jan. 1998), pp. 823–854. ISSN: 0143-1161. DOI: 10.1080/014311698215748. (Visited on 08/01/2026).

[70] Michael Schmitt and Xiao Xiang Zhu. “Data Fusion and Remote Sensing: An Ever-Growing Relationship”. In: *IEEE Geoscience and Remote Sensing Magazine* 4.4 (Dec. 2016), pp. 6–23. ISSN: 2168-6831. DOI: 10.1109/MGRS.2016.2561021. (Visited on 07/22/2026).

[71] Yi Wang et al. “SSL4EO-S12: A Large-Scale Multimodal, Multitemporal Dataset for Self-Supervised Learning in Earth Observation [Software and Data Sets]”. In: *IEEE Geoscience and Remote Sensing Magazine* 11.3 (Sept. 2023), pp. 98–106. ISSN: 2168-6831, 2473-2397, 2373-7468. DOI: 10.1109/MGRS.2023.3281651. (Visited on 07/16/2026).

[72] Yi Wang et al. *SSL4EO-S12: A Large-scale Multimodal Multitemporal Dataset for Self-supervised Learning in Earth Observation*. June 2022. DOI: 10.14459/2022MP1660427.001. (Visited on 07/16/2026).

[73] *Harmonized Sentinel-2 MSI: MultiSpectral Instrument, Level-2A (SR) | Earth Engine Data Catalog*. https://developers.google.com/earth-engine/datasets/catalog/COPERNICUS_S2_SR_HARMONIZED. (Visited on 07/29/2026).

[74] *Sentinel-1 SAR GRD: C-band Synthetic Aperture Radar Ground Range Detected, Log Scaling | Earth Engine Data Catalog*. https://developers.google.com/earth-engine/datasets/catalog/COPERNICUS_S1_GRD. (Visited on 07/27/2026).

[75] *Sentinel-1 Algorithms | Google Earth Engine*. https://developers.google.com/earth-engine/guides/sentinel1. (Visited on 07/27/2026).

[76] *Embed2scale/SSL4EO-S12-v1.1-Zarr · Datasets at Hugging Face*. https://huggingface.co/datasets/embed2scale/SSL4EO-S12-v1.1-Zarr. Feb. 2025. (Visited on 07/16/2026).

[77] *Sentinel-1 SAR GRD: C-band Synthetic Aperture Radar Ground Range Detected, Log Scaling | Earth Engine Data Catalog*. https://developers.google.com/earth-engine/datasets/catalog/COPERNICUS_S1_GRD. (Visited on 07/17/2026).

[78] Google. *Gemini 2.5 Flash-Lite | Gemini API*. https://ai.google.dev/gemini-api/docs/models/gemini-2.5-flash-lite?hl=de. (Visited on 09/29/2026).

[79] Thomas Wolf et al. "Transformers: State-of-the-Art Natural Language Processing". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Ed. by Qun Liu and David Schlangen. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6. (Visited on 08/19/2026).

[80] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc., 2019. (Visited on 08/19/2026).

[81] *Arya555/Vicuna-7b-v1.5-Hf · Hugging Face*. https://huggingface.co/arya555/vicuna-7b-v1.5-hf. July 2023. (Visited on 08/19/2026).

[82] Ruibin Xiong et al. "On Layer Normalization in the Transformer Architecture". In: *Proceedings of the 37th International Conference on Machine Learning*. PMLR, Nov. 2020, pp. 10524–10533. (Visited on 08/01/2026).

[83] Xavier Glorot and Yoshua Bengio. "Understanding the Difficulty of Training Deep Feedforward Neural Networks". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, Mar. 2010, pp. 249–256. (Visited on 08/01/2026).

[84] Nitish Srivastava et al. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15.56 (2014), pp. 1929–1958. ISSN: 1533-7928. (Visited on 08/01/2026).

[85] I. Loshchilov and Frank Hutter. "Decoupled Weight Decay Regularization". In: *International Conference on Learning Representations*. Nov. 2017. (Visited on 08/19/2026).

[86] Ilya Loshchilov and Frank Hutter. "SGDR: Stochastic Gradient Descent with Warm Restarts". In: *International Conference on Learning Representations*. Feb. 2017. (Visited on 06/21/2026).

[87] Takuya Akiba et al. "Optuna: A Next-generation Hyperparameter Optimization Framework". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '19. New York, NY, USA: Association for Computing Machinery, July 2019, pp. 2623–2631. ISBN: 978-1-4503-6201-6. DOI: 10.1145/3292500.3330701. (Visited on 09/02/2026).

[88] James Bergstra et al. “Algorithms for Hyper-Parameter Optimization”. In: *Advances in Neural Information Processing Systems*. Vol. 24. Curran Associates, Inc., 2011. (Visited on 09/02/2026).

[89] Kishore Papineni et al. “Bleu: A Method for Automatic Evaluation of Machine Translation”. In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Ed. by Pierre Isabelle, Eugene Charniak, and Dekang Lin. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, July 2002, pp. 311–318. DOI: 10.3115/1073083.1073135. (Visited on 09/02/2026).

[90] Boxing Chen and Colin Cherry. “A Systematic Comparison of Smoothing Techniques for Sentence-Level BLEU”. In: *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Ed. by Ondřej Bojar et al. Baltimore, Maryland, USA: Association for Computational Linguistics, June 2014, pp. 362–367. DOI: 10.3115/v1/W14-3346. (Visited on 09/02/2026).

[91] Matt Post. “A Call for Clarity in Reporting BLEU Scores”. In: *Proceedings of the Third Conference on Machine Translation: Research Papers*. Ed. by Ondřej Bojar et al. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 186–191. DOI: 10.18653/v1/W18-6319. (Visited on 09/02/2026).

[92] Chin-Yew Lin. “ROUGE: A Package for Automatic Evaluation of Summaries”. In: *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 74–81. (Visited on 09/02/2026).

[93] *Evaluate · Hugging Face*. https://huggingface.co/docs/evaluate/v0.4.5/index. (Visited on 09/11/2026).

[94] Satanjeev Banerjee and Alon Lavie. “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments”. In: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Ed. by Jade Goldstein et al. Ann Arbor, Michigan: Association for Computational Linguistics, June 2005, pp. 65–72. (Visited on 09/02/2026).

[95] George A. Miller. “WordNet: A Lexical Database for English”. In: *Commun. ACM* 38.11 (Nov. 1995), pp. 39–41. ISSN: 0001-0782. DOI: 10.1145/219717.219748. (Visited on 09/02/2026).

[96] Nils Reimers and Iryna Gurevych. “Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Ed. by Kentaro Inui et al. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3982–3992. DOI: 10.18653/v1/D19-1410. (Visited on 09/02/2026).

[97] *Sentence-Transformers/All-Mpnet-Base-v2 · Hugging Face*. https://huggingface.co/sentence-transformers/all-mpnet-base-v2. Jan. 2024. (Visited on 09/02/2026).

[98] *NLTK :: Natural Language Toolkit*. https://www.nltk.org/. (Visited on 09/11/2026).

[99] M. Hodosh, P. Young, and J. Hockenmaier. "Framing Image Description as a Ranking Task: Data, Models and Evaluation Metrics". In: *Journal of Artificial Intelligence Research* 47 (Aug. 2013), pp. 853–899. ISSN: 1076-9757. DOI: 10.1613/jair.3994. (Visited on 09/09/2026).

[100] Zhenghang Yuan et al. "ChatEarthNet: A Global-Scale Image-Text Dataset Empowering Vision-Language Geo-Foundation Models". In: *Earth System Science Data* 17.3 (Mar. 2025), pp. 1245–1263. ISSN: 18663516. DOI: 10.5194/essd-17-1245-2025. arXiv: 2402.11325.

[101] Zhiming Ma et al. *SARChat-Bench-2M: A Multi-Task Vision-Language Benchmark for SAR Image Interpretation*. Mar. 2025. DOI: 10.48550/arXiv.2502.08168. arXiv: 2502.08168 [cs.CL]. (Visited on 09/03/2026).

[102] Fabio Montello, Edoardo Arnaudo, and Claudio Rossi. "MMFlood: A Multimodal Dataset for Flood Delineation From Satellite Imagery". In: *IEEE Access* 10 (2022), pp. 96774–96787. ISSN: 21693536. DOI: 10.1109/ACCESS.2022.3205419.

[103] Hyeongkyun Kim and Orestis Oikonomou. "ZeroFlood: Flood Hazard Mapping from Single-Modality SAR Using Geo-Foundation Models". In: *Proceedings of the European Conference on Synthetic Aperture Radar, EUSAR*. Ed. by Grzegorz Adamiuk and Martin Stangl. Baden-Baden, Germany: VDE, June 2026. (Visited on 09/03/2026).

A Prompting

A.1 SAR-text Dataset Creation

The creation of the SAR-text dataset has already been explained in Section 3.3. The exact model was Gemini 2.5 Flash-Lite and used with an API. For the instruction, the following prompt has been used.

```
[MISSION & PHILOSOPHY]
You are a Senior SAR Remote Sensing Scientist. Your task is to
↪  provide a physical and structural analysis of a Sentinel-1
↪  image (TerraMesh RGB).
You MUST systematically search for both HUMAN/GEOMETRIC features
↪  and PHYSICAL NATURAL cover. Do NOT ignore localized
↪  settlements or corner details.

[STEP 1: GEOMETRIC & STRUCTURAL DETECTION]
Scan the entire frame (including margins and corners) for
↪  ENGINEERED SIGNATURES:
- CLUSTERED BRIGHT WHITE/GOLD PINPOINTS: These are structural
↪  "Double-Bounce" reflections from metallic/concrete buildings,
↪  isolated houses, or small villages. EVEN IF LOCATED IN A
↪  CORNER OR SMALL IN SIZE, REPORT THEM AS A SETTLEMENT OR
↪  FARMSTEAD.
- RECTANGULAR / PARALLEL BOUNDARIES: Indicates engineered spatial
↪  management (agricultural fields or urban plots).
    * IF RECTANGULAR + TEXTURED GOLD/GREEN: Managed cultivated
    ↪  field or active crops.
    * IF RECTANGULAR + SMOOTH CYAN/BLACK: Tilled bare soil, paved
    ↪  plaza, or field with no crop structure.
- OVAL / CIRCULAR SMOOTH SHAPES: Stadiums, arenas, or circular
↪  storage tanks.

[STEP 2: PHYSICAL NATURAL COVER (SAR PHYSICS)]
- GRAINY VIBRANT GOLD / ORANGE: High-volume 3D canopy biomass
↪  (Dense forests or mature woodlands).
- SOFT GREENISH / DULL YELLOW: Low-biomass ground cover (Pastures,
↪  grasslands, or low vegetation).
- DESATURATED YELLOW WITH BLUE HAZE: Exposed bare soil or tilled
↪  agricultural ground.
- SINUOUS DARK BLUE / BLACK: Natural river systems or streams
↪  (Specular reflection).
```

```
[STEP 3: MANDATORY DUAL-OUTPUT CHECK]
Before writing, verify:
1. Did I search for bright structural pinpoints in the
 ↪  corners/edges? (e.g., the bright cluster in the lower corner
 ↪  represents a rural settlement).
2. Did I contrast the 3D canopy biomass with agricultural or built
 ↪  surfaces?

[TASK & VOCABULARY MANDATE]
Generate a professional geographic caption (3-5 sentences).
- Start directly with the main landscape type (e.g., "A mixed
 ↪  rural landscape of dense woodland and cultivated fields...").
- Explicitly report localized settlements or building clusters
 ↪  detected via high-intensity double-bounce pinpoints (do not
 ↪  skip corner settlements).
- Use physical descriptors: "3D canopy biomass", "double-bounce
 ↪  structural reflections", "managed geometric fields", "sinuous
 ↪  waterway".
- DO NOT use technical jargon like 'dB', 'VH', 'VV', or
 ↪  'backscatter'.

[STYLE EXAMPLE]
A mixed rural environment is defined by dense patches of
 ↪  high-volume 3D canopy biomass interspersed with geometric
 ↪  cultivated fields. In the lower corner, a concentrated cluster
 ↪  of intense white double-bounce reflections reveals a localized
 ↪  rural settlement connected by minor access corridors. A
 ↪  sinuous dark waterway cuts through the forested terrain,
 ↪  contrasting with the smooth, tilled surfaces of adjacent
 ↪  agricultural plots.
```

A.2 SAR-specific Prompt for Decoder

The decoder never receives an image directly. The placeholder shown below indicates the position where the bridge inserts its projected patch embeddings into the token sequence. These embeddings are placed between the embedded prefix and suffix. The prompt therefore does not describe how visual information is encoded, but only provides the context for interpreting the visual information and specifies the expected output format. The string ends with a single blank after the assistant marker.

```
"The assistant is a remote sensing scientist and reads Sentinel-1
 ↪  radar observations in terms of physical scattering:
 ↪  double-bounce, volume, rough-surface and specular returns,
 ↪  geometric boundaries and linear low-return paths. USER:
 ↪  <visual tokens>
Describe this scene in 3-5 sentences in a professional
 ↪  geographic-analysis tone. Consider the whole scene, including
 ↪  edge and corner regions, and tie every statement to a
 ↪  scattering signature you can actually observe. Avoid acronyms,
 ↪  and do not open with a fixed template phrase. ASSISTANT: "
```

The prefix and suffix are stored separately and cached in the same way as the default prompt. This allows the prompt to be changed without modifying the sequence construction, loss calculation, or generation process. For the Sentinel-2 experiments, the default prompt from Section 4.3 is used.

A.3 Prompts for Baseline

The baseline uses the same frozen language model, but without the geospatial encoder and the bridge. Instead, it receives a rendered three-channel image through its own vision tower. The placeholder in the prompt blocks below therefore represents the image tokens generated by this vision tower. The wording of the prompts is kept as close as possible to the prompts used for the decoder, so that the comparison focuses on the different visual pathways and not on differences in the instructions.

For the optical track, the two prompts are identical character for character. The baseline therefore receives exactly the same instruction as the bridge-based system.

```
"USER: <image tokens>
Describe the surface features shown in this satellite observation.
 ↪  ASSISTANT: "
```

In the SAR track, one difference between the two prompts is necessary because the visual inputs are represented differently. The bridge receives the calibrated dual-polarisation channels directly, while the baseline receives a false-colour image. In this rendering, the cross-polarised return is assigned to the red channel, the co-polarised return to the green channel, and the difference between the two to the blue channel. Without this information, the model could interpret the colours as properties of an optical image and describe the observed hues instead of the underlying scattering behaviour. One additional sentence is therefore included in the prefix. This is the only difference from the decoder prompt defined in Section A.2.

```
"The assistant is a remote sensing scientist and reads Sentinel-1
 ↪  radar observations in terms of physical scattering:
 ↪  double-bounce, volume, rough-surface and specular returns,
 ↪  geometric boundaries and linear low-return paths. The
 ↪  observation is shown as a false-colour composite of the two
 ↪  polarisation channels: red is VH backscatter, green is VV
 ↪  backscatter, and blue is the VV-VH difference, each stretched
 ↪  from a fixed decibel range. USER: <image tokens>
Describe this scene in 3-5 sentences in a professional
 ↪  geographic-analysis tone. Consider the whole scene, including
 ↪  edge and corner regions, and tie every statement to a
 ↪  scattering signature you can actually observe. Avoid acronyms,
 ↪  and do not open with a fixed template phrase. ASSISTANT: "
```

That added sentence has to be acknowledged when the two radar arms are compared. It hands the baseline an explicit description of its own input representation, which the bridge never receives. The radar comparison is therefore mildly generous towards the baseline, and any gap that remains can be read as a lower bound on what the encoder contributes.

B Training Details

This appendix gives an overview of the bridge training procedure and the quantities used to monitor its behaviour during training. Since both the encoder and the language model remain frozen, the optimisation is restricted to the bridge. The purpose of the training is therefore to determine whether the representations produced by the pretrained encoder can be transformed in a way that makes them usable by the language model. The diagnostic introduced in Section B.4 is used to examine this process.

B.1 Fine-tuning

The learning rate was the only training parameter that was determined experimentally. The remaining settings were fixed by the experimental design or were taken directly from the reference configuration and kept unchanged.

The selection was carried out in two stages. First, a small manual sweep on both modalities was used to determine a suitable range. For the MLP bridge, three runs were performed with learning rates of 1×10^{-5}, 3×10^{-5}, and 1×10^{-4}. The linear bridge was tested at 3×10^{-5}, 1×10^{-4}, and 3×10^{-4}. In all runs, the minimum learning rate was set to one tenth of the initial value, so the relative shape of the learning-rate schedule remained the same. In addition, two runs tested weight decay values of zero and 1×10^{-4}. One further run used a reduced amount of training data to inspect the resulting learning curve.

The purpose of this first stage, which is visualised in Figure B.1 was to identify the relevant range for the later search. The results indicated that the two bridge variants work best at different learning-rate ranges, with the linear bridge requiring a higher rate than the MLP bridge. The tests with weight decay also showed only a small effect at the given number of trainable parameters. These observations were then used to define the search intervals for the automated stage. This is why the ranges reported in Table B.1 are different for the two bridge variants. The tests also showed that ten epochs are sufficient to find the best training state at a suitable learning rate.

B.2 Configuration files

The configuration setup described in Section 5.1.4 is also used for the sweeps. Each Optuna trial resolves and stores its own configuration in the corresponding trial

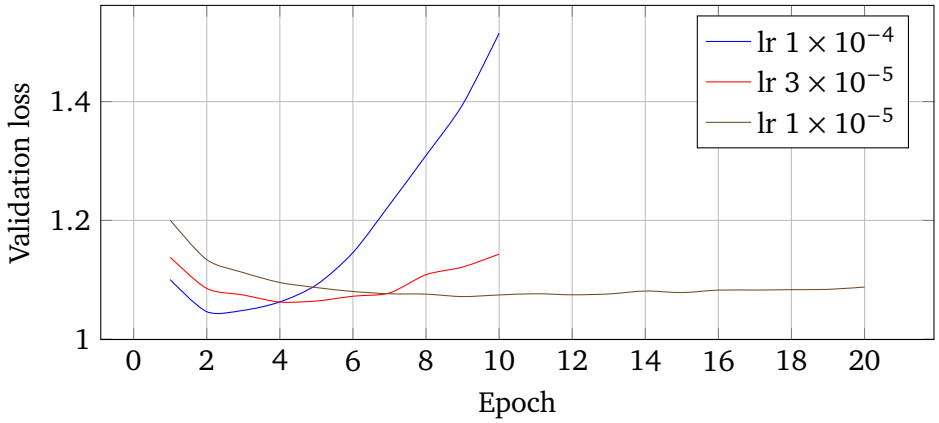


Figure B.1: Manual stage on Sentinel-2 with the MLP bridge at three learning rates.

directory. This makes it possible to repeat an individual trial with a normal training call without running the complete study again. The structure of the configuration files is shown below.

```
configs/
├── finetuning/ ...  manual runs and the sweep definitions
│   ├── PT_lr_*.yaml ...  manual grid over the learning rate
│   ├── PT_weight_decay_*.yaml ...  weight decay
│   └── optuna_*.yaml ...  one file per study
└── baseline/ ...  zero-shot baseline per modality
```

B.3 Learning-rate sweeps

Within each study, the bridge architecture, modality, decoder, and prompt remain unchanged. Only the parameters included in the respective search are allowed to vary. This makes it possible to compare the best result of one study with the best result of another without changing several design choices at the same time. The individual studies are listed in Table B.1.

Two pruning rules can stop a trial before the ten scheduled epochs are completed and they use different reference points. The median pruner compares a trial with the other trials in the same study. It becomes active after five completed trials and five reported epochs and stops a trial if its validation loss is worse than the median of the other trials. The second rule compares the current validation loss with the best value reached by the same trial. From the third epoch onwards, the trial is stopped if its validation loss is more than five percent above its own best value. This avoids stopping trials while the learning-rate schedule is still in its warmup phase. All winning runs in Table B.2 were stopped by the second rule after five to seven of the ten scheduled

epochs. Further details on the pruning parameters and their effect on the training duration are provided in Section B.3.

Table B.1: The sweeps conditions. The rate is sampled log-uniformly inside the given range.

Study	Bridge	Modality	Decoder	Prompt	Rate range
`mlp_s1`	MLP	S1GRD	LLaVA	default	1×10^{-5} to 3×10^{-4}
`mlp_s1_prompt`	MLP	S1GRD	LLaVA	sar	1×10^{-5} to 3×10^{-4}
`mlp_s1_vicuna`	MLP	S1GRD	Vicuna	default	1×10^{-5} to 3×10^{-4}
`mlp_s2`	MLP	S2L2A	LLaVA	default	1×10^{-5} to 3×10^{-4}
`mlp_s2_vicuna`	MLP	S2L2A	Vicuna	default	1×10^{-5} to 3×10^{-4}
`linear_s2`	Linear	S2L2A	LLaVA	default	3×10^{-5} to 1×10^{-3}
`linear_s1`	Linear	S1GRD	LLaVA	default	3×10^{-5} to 1×10^{-3}

Table B.2: Winning trial per study, selected on validation loss.

Study	Trial	Learning rate	Validation loss
`mlp_s1`	9	2.537×10^{-4}	0.9975
`mlp_s1_prompt`	12	2.794×10^{-4}	1.0035
`mlp_s1_vicuna`	3	2.537×10^{-4}	1.0173
`mlp_s2`	12	2.712×10^{-4}	1.0229
`mlp_s2_vicuna`	3	2.537×10^{-4}	1.0523
`linear_s1`	9	8.413×10^{-4}	1.0535
`linear_s2`	12	8.259×10^{-4}	1.0414

An important point in Table B.2 is the location of the selected learning rates within the search range. All five selected values are between 2.5×10^{-4} and 2.8×10^{-4}, while the upper limit of the search interval is 3×10^{-4}. Since the best values are close to this upper boundary, the results do not necessarily show that the optimum learning rate has been found. Higher learning rates could therefore still result in a lower validation loss. Due to computing limitations on the cluster, no further runs could have been performed. The validation loss per study in the winner run is visualised in Figure B.2.

The reported values should consequently be interpreted as suitable learning rates within the tested range, but not as confirmed optima. An extension of this work should first repeat the search with a higher upper limit to determine whether the validation loss continues to improve beyond 3×10^{-4}.

B.4 Grounding gap

A low caption loss does not by itself show that the model uses the information from the corresponding scene. Since captions within a dataset can contain similar descriptions and recurring phrases, some tokens can be predicted from the language model's learned patterns without requiring the correct visual input. The grounding gap is

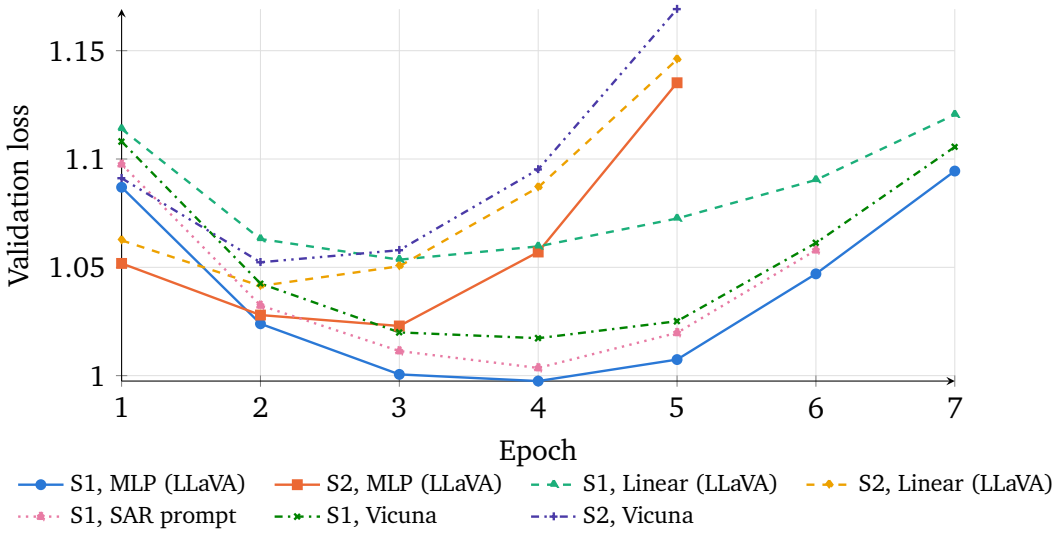


Figure B.2: Validation loss of the winning trial of each study.

used to examine how the validation loss changes when the correspondence between scenes and captions is removed.

For each validation epoch, two passes are performed. The first pass uses the validation batch in its original form. For the second pass, the same projected visual tokens are shifted by four positions within the batch before they are passed to the decoder. The captions remain unchanged, so each caption is evaluated together with the visual representation of another scene. The set of visual inputs and captions is therefore the same in both passes, but their correspondence is different.

$$g_e = L_e^{\text{shuffled}} - L_e^{\text{val}} \tag{B.1}$$

A value close to zero means that changing the visual input has little effect on the validation loss. In this case, the caption predictions can be made with similar loss even when the scene does not correspond to the caption. A larger positive gap means that the loss increases when the visual and textual inputs are mismatched, which provides evidence that the projected visual information contributes to the caption prediction. The magnitude of the gap expresses this difference in terms of the loss. The shift requires at least two samples in a batch. Validation batches with fewer samples are therefore skipped for this comparison.

The resulting curves were inspected manually and no automated threshold was applied. The purpose of this inspection was to identify runs in which the visual input appeared to have no measurable effect on the loss. Such a result would have been a reason to exclude the corresponding study because it would suggest that the captions could be generated without using the projected visual features. None of the studies showed this behaviour, so no study was excluded based on the grounding gap.

The grounding gap was not used for selecting the reported study or determining the training duration. Both decisions were based on the validation loss. The grounding gap therefore served only as an additional diagnostic for checking whether the visual input had an observable influence on the decoder.

C Computation

All experiments presented in this thesis were executed as batch jobs on a shared computing cluster, with a log file created for each job. The computational cost of the experiments is taken from these logs. This appendix only includes the runs used for the reported results, while the earlier exploratory experiments are not included.

C.1 Hardware and software

All experiments were executed on the LRZ HPDA terrabyte computing infrastructure. Each job was assigned one NVIDIA A100-SXM4-80GB GPU, which was used exclusively by that job. The pipeline does not use data parallelism or model parallelism, so all reported wall-clock times correspond to a single GPU. They can therefore be compared directly with reproductions using one GPU. The jobs were scheduled across several nodes of the same partition, so they were not restricted to a specific machine.

A completed run with a batch size of 16 reached a peak memory usage of 73.2 GiB out of the available 80 GiB. The average GPU utilisation during the run was 83 %, with a maximum of 100 %. Increasing the batch size to 32 was not possible. This run terminated with a CUDA out-of-memory error after the memory usage reached 78.7 GiB.

The batch size of 16 used for all reported experiments was therefore chosen based on the available GPU memory and not as a separately optimised training parameter. It is the largest batch size that fits reliably on an 80 GiB GPU for this configuration. Since the observed peak usage leaves less than 2 GiB of available memory, the GPU needs to be used without memory being shared with another process. Systems with less than 80 GiB of GPU memory would require a smaller batch size.

C.2 Computational Cost

C.2.1 Training Cost

All trials use the same data and training settings. Each trial is trained on 15,360 acquisitions and validated on 2560 acquisitions using a batch size of 16 and ten scheduled epochs. Since these settings are identical for all trials, the duration of one epoch provides a useful basis for estimating the computational cost. The measured

epoch times are relatively consistent across the runs. Table C.1 summarises the cost of an epoch in each study.

Table C.1: Cost of the seven learning-rate sweeps, counted over the epochs of the recorded trials. The epoch time is the median over those epochs.

Study	Time per epoch
`mlp_s1`	21.9 min
`mlp_s1_prompt_sar`	22.0 min
`mlp_s1_vicuna`	21.9 min
`linear_s1`	22.0 min
`mlp_s2`	19.4 min
`mlp_s2_vicuna`	22.5 min
`linear_s2`	18.9 min

Considering all epochs, the median runtime is 21.9 min. The choice of bridge architecture and decoder has only a small effect on the epoch duration. This is expected because the encoder and decoder remain frozen during training, while the trainable bridge contains only a small part of the parameters involved in the forward and backward passes.

The two Sentinel-2 studies using the LLaVA decoder have the lowest epoch times, with values of around 19 min. The Sentinel-1 studies are consistently closer to 22 min.

C.2.2 Inference Cost

After the best trial was selected, its model was evaluated on the twenty test batches of the respective study. Together, these batches cover 1,280 scenes, with four samples generated for each scene. This results in 5,120 generated captions for each study. The maximum number of generated tokens differs between the two tracks, with a budget of 200 tokens for Sentinel-1 and 100 tokens for Sentinel-2 (Section 5.1.2).

The resulting runtimes are shown in Table C.2. Since the decoder generates the captions autoregressively, longer outputs require more decoding steps and therefore increase the inference time. The larger token budget for Sentinel-1 consequently results in an inference time that is roughly three times higher. The generated captions also support this explanation. The median caption length is 192 characters for Sentinel-2 and 672 characters for Sentinel-1, giving a similar ratio to the measured inference times.

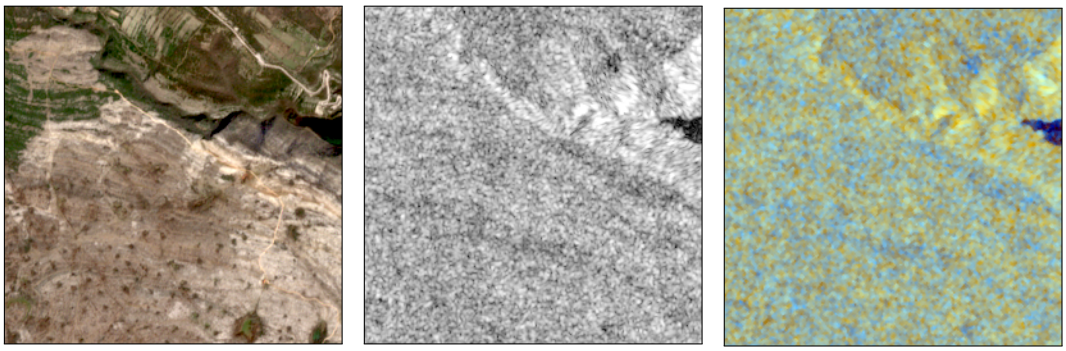
Table C.2: Inference over the winning trials, 5120 generations per study. The time per generation is the mean interval between consecutive generations.

Study	Budget	Caption length	Per generation	Per study
`mlp_s1`	200 tokens	672 characters	3.6 s	307 min
`linear_s1`	200 tokens	652 characters	3.5 s	300 min
`mlp_s1_prompt_sar`	200 tokens	721 characters	3.8 s	327 min
`mlp_s1_vicuna`	200 tokens	672 characters	3.6 s	304 min
`mlp_s2`	100 tokens	192 characters	1.3 s	107 min
`linear_s2`	100 tokens	195 characters	1.3 s	105 min
`mlp_s2_vicuna`	100 tokens	196 characters	1.3 s	107 min
`baseline_llava_s1`	200 tokens	598 characters	3.4 s	288 min
`baseline_llava_s2`	100 tokens	364 characters	2.5 s	212 min

D Additional Examples of the Results

This appendix contains additional examples from the test split. Each row shows the Sentinel-2 true-colour image, the Sentinel-1 VH backscatter, and the false-colour composite used for SAR caption generation. The reference captions and the captions generated by the base configuration for each track are shown below the images. All images and visualized from the SSL4EO-S12 v1.1 dataset [41] and the reference S2 captions are taken from the Llama3-SSL4EO-S12 dataset [52].

S2 L2A (RGB) **S1 GRD (VH)** **S1 false-colour composite**



S2 reference caption (Llama3-SSL4EO-S12)

The image is a satellite view of a region that includes the Krka National Park and the Rit wetland area. The landscape is a mix of natural and agricultural land uses, with a significant portion of the land dedicated to agriculture and a smaller portion to natural areas such as forests and meadows. The forested area is located adjacent to the meadow, with the meadow extending to the right of the forested area.

S2 generated caption (base configuration)

The image is a satellite view of a landscape with various land uses, including agricultural fields and natural areas. There are also some water features visible in the image.

S1 reference caption (Gemini)

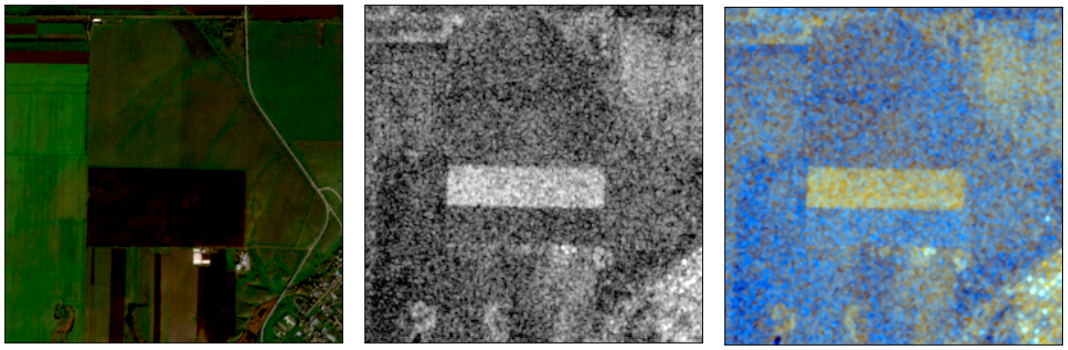
This landscape is dominated by expansive, low-biomass ground cover and grasslands, characterized by a uniform, soft greenish-yellow texture. A prominent band of high-volume 3D canopy biomass stretches across the upper portion of the frame, showing the vibrant, grainy signature of dense woody vegetation. Within the upper right quadrant, a localized cluster of intense white double-bounce structural reflections indicates a small, engineered settlement or farmstead complex nestled against the forest edge. Adjacent to this structure, a distinct, dark, non-reflective feature suggests a small body of water or shadowed depression, contrasting sharply with the surrounding managed vegetation and open fields.

S1 generated caption (base configuration)

This landscape is dominated by high-volume 3D canopy biomass, characterized by a vibrant golden texture indicative of dense woodland or mature vegetation. A distinct dark blue specular feature in the upper right corner suggests a localized water body or small basin that contrasts with the surrounding terrain. In the lower margin and scattered throughout the frame are subtle geometric boundaries suggesting managed agricultural plots within this natural environment. While the area appears largely undisturbed, careful inspection reveals faint, isolated bright pinpoints representing minor structural double-bounce reflections from small rural outbuildings or farmstead infrastructure embedded within the forest cover.

Figure D.1: Scene 35129, acquisition time 2.

S2 L2A (RGB) | **S1 GRD (VH)** | **S1 false-colour composite**



S2 reference caption (Llama3-SSL4EO-S12)

The image is a satellite view of a large agricultural area with various land uses, including farmland, grassland, and a meadow. There is a large rectangular brown area, possibly a farmyard, and a dark blue reservoir. The image also shows a residential area, a cemetery, and a road.

S2 generated caption (base configuration)

The image is a satellite view of an agricultural area with various crops, including green and brown fields. There are also residential areas visible in the background.

S1 reference caption (Gemini)

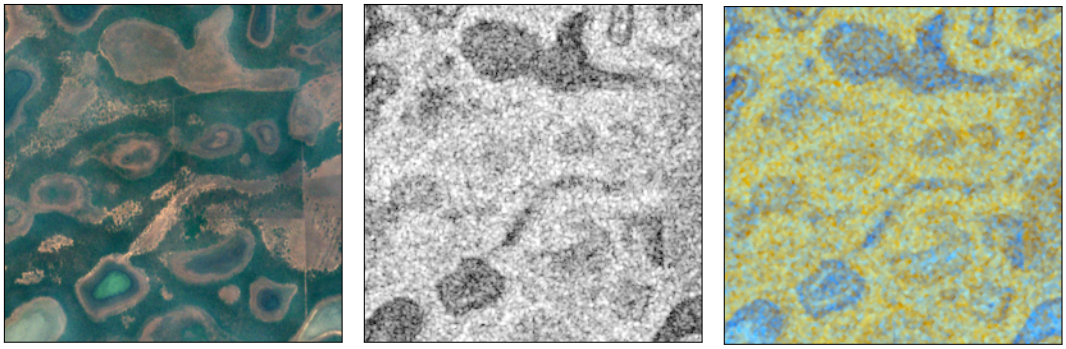
This landscape is characterized by a mix of low-biomass ground cover and managed geometric land use. A distinct rectangular plot, showing a muted golden texture, stands out against the surrounding blue-toned, low-lying terrain. In the lower-right corner, a cluster of high-intensity double-bounce structural reflections indicates the presence of a localized farmstead or small built facility. This composition effectively demonstrates the contrast between natural, smooth-surfaced ground and the engineered, angular features of rural development.

S1 generated caption (base configuration)

This agricultural landscape is characterized by a mosaic of managed geometric fields exhibiting varying degrees of soil moisture and vegetation. The scene features distinct rectangular plots with smooth, desaturated surfaces indicative of tilled or fallow ground, contrasting against the textured gold-toned patches representing active crop growth. In the lower right corner, a cluster of intense white double-bounce structural reflections indicates a localized farmstead or small building complex integrated into the field grid. These engineered signatures highlight an area where human land management has been applied to optimize spatial organization within the rural environment.

Figure D.2: Scene 231325, acquisition time 3.

S2 L2A (RGB) **S1 GRD (VH)** **S1 false-colour composite**



S2 reference caption (Llama3-SSL4EO-S12)

A satellite image of a large, irregularly shaped island with a flat top and a few small protrusions, surrounded by five smaller landmasses. The image has a muted color palette with shades of blue, green, and gray.

S2 generated caption (base configuration)

The image is a satellite view of a landscape with various land cover types, including grassland and wetlands. There are several bodies of water visible in the image, such as ponds and lakes.

S1 reference caption (Gemini)

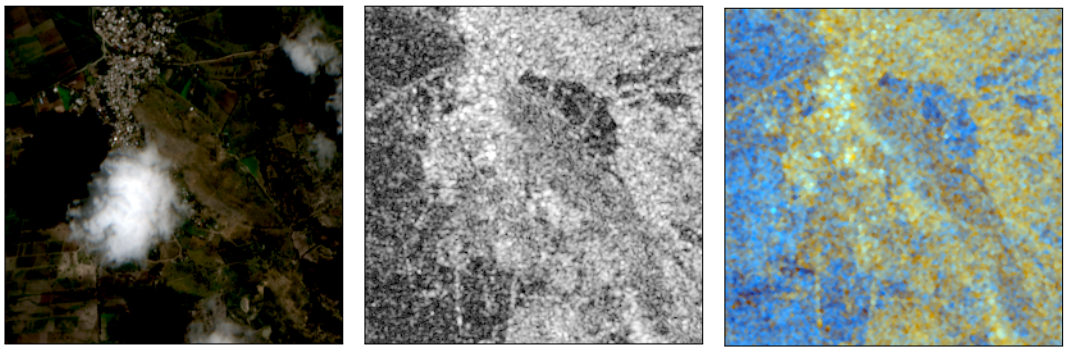
This landscape is dominated by expansive, undulating patches of high-volume 3D canopy biomass, characterized by a vibrant gold and orange texture indicative of mature woodland or dense shrubland. Interspersed throughout this natural cover are distinct, desaturated blue-grey zones representing low-lying areas or seasonal floodplains where smooth specular reflection suggests saturated ground or standing water. A thorough scan of the frame reveals no significant double-bounce structural reflections, indicating a lack of human-engineered settlements or rigid geometric agricultural management within this specific domain. The terrain is defined by the interplay between the dense, coarse-textured forest cover and these softer, darker moisture-retaining depressions.

S1 generated caption (base configuration)

This landscape is dominated by high-volume 3D canopy biomass, characterized by a vibrant golden texture indicative of dense, mature woodland or scrub vegetation. Interspersed throughout the frame are patches of soft greenish ground cover and desaturated blue areas representing low-biomass grasslands or exposed soil surfaces. While there is no significant evidence of engineered geometric fields within this specific view, subtle localized clusters of bright pinpoints suggest minor structural reflections from isolated natural features or small rural infrastructure. The overall composition reflects an undisturbed natural environment with minimal human intervention in the immediate area.

Figure D.3: Scene 20193, acquisition time 3.

S2 L2A (RGB) | **S1 GRD (VH)** | **S1 false-colour composite**



S2 reference caption (Llama3-SSL4EO-S12)

A satellite image showing a large, fluffy white cloud, a cluster of buildings, and a rocky terrain, with green fields in the background.

S2 generated caption (base configuration)

The image is a satellite view of a landscape that includes agricultural fields, residential areas, and natural features such as forests. There are also clouds visible in the sky.

S1 reference caption (Gemini)

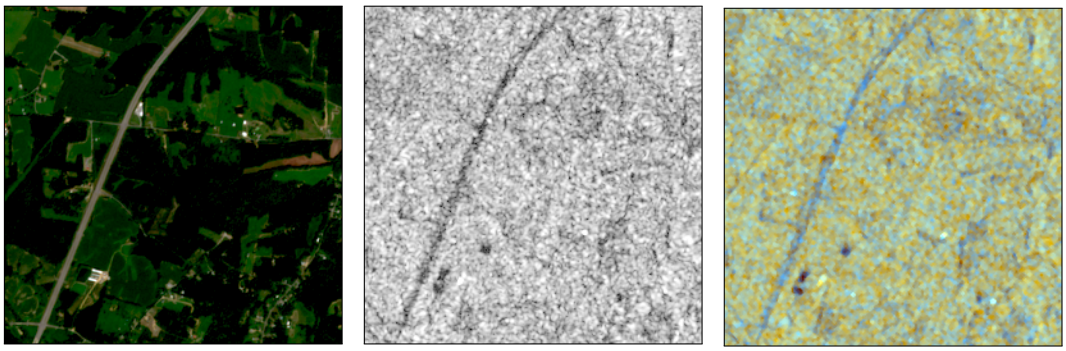
This landscape is dominated by dense, high-volume 3D canopy biomass, characterized by vibrant golden textures representing mature woodland or scrub vegetation. Interspersed within the forested terrain are patches of desaturated blue, indicating moisture-rich ground or open areas with low vegetation cover. Notably, several clusters of intense white double-bounce structural reflections are visible, particularly concentrated toward the upper and lower left margins, marking localized man-made dwellings or small-scale rural infrastructure. These geometric signals break the natural heterogeneity of the foliage, suggesting human presence integrated into the dense natural canopy.

S1 generated caption (base configuration)

This landscape is characterized by a mosaic of high-volume 3D canopy biomass and lower-lying, moisture-rich vegetation. The terrain displays a distinct transition between vibrant gold textures representing dense woodland cover and desaturated blue regions indicative of exposed soil or low ground moisture levels. Within the upper left quadrant, a localized cluster of intense white double-bounce structural reflections signifies a small rural settlement or isolated farmstead complex. These engineered pinpoints contrast sharply with the surrounding natural topography, highlighting an area where human activity intersects with undeveloped wilderness.

Figure D.4: Scene 145981, acquisition time 0.

S2 L2A (RGB) **S1 GRD (VH)** **S1 false-colour composite**



S2 reference caption (Llama3-SSL4EO-S12)

The image is a satellite view of a landscape with a road, two cemeteries, and green grass.

S2 generated caption (base configuration)

The image is a satellite view of a landscape with two roads, one running horizontally and the other vertically. There are also several patches of green land and brown land in different areas of the image.

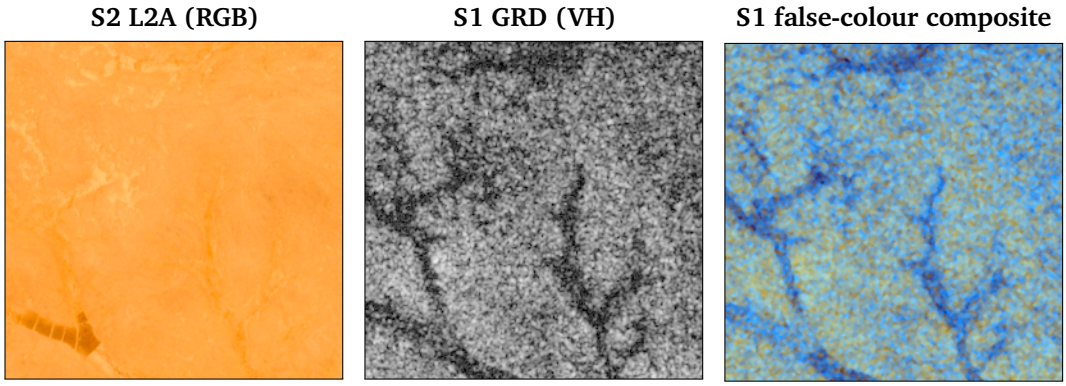
S1 reference caption (Gemini)

A landscape dominated by dense, high-volume 3D canopy biomass covers the majority of the frame, characterized by a granular gold and orange texture. A prominent, sinuous dark blue line traverses the terrain, representing a natural waterway or drainage system with low surface roughness. Careful inspection of the lower edges reveals sporadic, high-intensity white pinpoints, indicating localized double-bounce structural reflections from small-scale human settlements or isolated infrastructure. The overall environment reflects a largely undisturbed natural ecosystem interspersed with minimal, site-specific anthropogenic footprint.

S1 generated caption (base configuration)

This landscape is dominated by high-volume 3D canopy biomass, characterized by a vibrant golden texture indicative of mature woodland or dense scrub. A distinct linear feature cuts across the upper portion of the frame, representing an engineered corridor or path that contrasts with the surrounding natural vegetation. In the lower left corner and scattered throughout the periphery, localized clusters of intense white double-bounce structural reflections indicate small farmsteads or isolated rural dwellings embedded within the forest cover. The overall terrain displays minimal geometric management, suggesting a largely undisturbed environment where human presence remains limited to these sparse, pinpoint settlements.

Figure D.5: Scene 224582, acquisition time 0.



S2 reference caption (Llama3-SSL4EO-S12)

The image is a satellite view of a brown, irregularly shaped landmass with jagged edges and varying elevations. There are no visible bodies of water in the image.

S2 generated caption (base configuration)

The image shows a brown, rough-textured land surface with no visible water features.

S1 reference caption (Gemini)

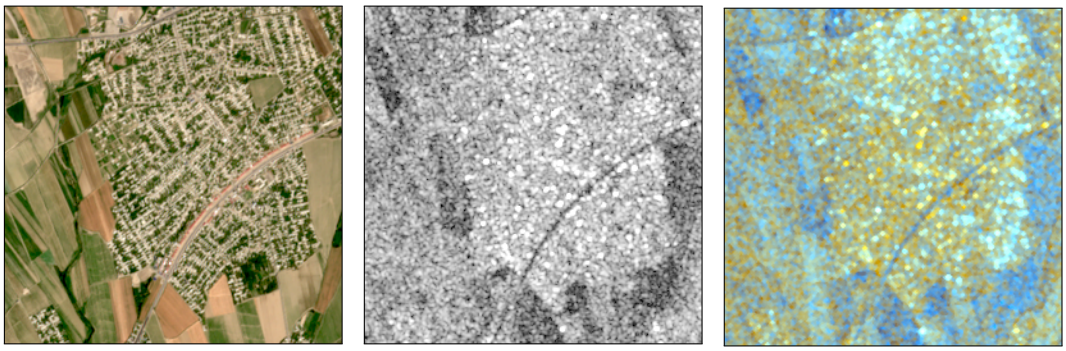
This landscape is characterized by a natural, low-biomass environment dominated by soft greenish ground cover and sparse vegetation. A series of sinuous dark blue channels traverse the terrain, indicating a natural drainage or stream system with significant specular reflection. There is a distinct absence of high-intensity double-bounce structural reflections, suggesting that this area is devoid of engineered settlements, building clusters, or managed geometric fields. The overall texture indicates a consistent, uniform expanse of wild terrain rather than an anthropogenically modified landscape.

S1 generated caption (base configuration)

This landscape is characterized by a mosaic of low-biomass ground cover and expansive, desaturated bare soil patches. The terrain features subtle variations in texture, with the vibrant gold tones indicating areas of moderate 3D canopy biomass or sparse vegetation. A prominent sinuous dark waterway cuts diagonally across the frame, creating distinct specular reflections that contrast sharply against the surrounding natural environment. While there are no significant clusters of intense white double-bounce structural reflections indicative of human settlements within this specific field of view, the lack of geometric agricultural boundaries suggests an undeveloped or naturally managed rural area.

Figure D.6: Scene 111330, acquisition time 3.

S2 L2A (RGB) **S1 GRD (VH)** **S1 false-colour composite**



S2 reference caption (Llama3-SSL4EO-S12)

The image is a satellite view of a city with various layers of land use, including vegetation, buildings, and bare ground. The topmost layer is green, the bottommost layer is brown, and the middle layer is white.

S2 generated caption (base configuration)

The image is a satellite view of a city with various land uses, including residential areas, agricultural fields, and industrial zones. There are also roads and highways visible in the image.

S1 reference caption (Gemini)

This landscape is characterized by a mosaic of dense, high-volume 3D canopy biomass and low-lying, desaturated ground cover. Scattered throughout the frame, particularly evident in the upper and central sectors, are clusters of vibrant white double-bounce structural reflections that signify isolated farmsteads or localized settlement infrastructure. The terrain is partitioned by subtle geometric boundaries, separating the active vegetation from smoother, cyan-toned regions of bare or tilled soil. A faint, linear path suggests minor human-made access corridors traversing the dense patches of golden canopy, providing contrast to the otherwise natural woodland environment.

S1 generated caption (base configuration)

This landscape is characterized by a high-density mosaic of textured gold and cyan vegetation, indicating an active agricultural or pastoral environment. Interspersed throughout the frame are distinct geometric boundaries that suggest managed land use, including patches of desaturated yellow bare soil and smooth blue-hued surfaces indicative of tilled ground. In the upper right corner, a concentrated cluster of intense white double-bounce structural reflections signifies a localized rural settlement or farmstead complex. The overall composition highlights a highly engineered terrain where natural biomass is interwoven with human spatial management.

Figure D.7: Scene 94762, acquisition time 3.

Use of AI-Based Tools

The following table lists where AI-based tools were used in the preparation of this thesis. All AI-generated suggestions were reviewed, adapted, and verified by the author, who takes full responsibility for the content.

Area	**Type of support**	**Tool**
Coding		
Plotting	Implementation and refinement of figures	Anthropic Claude
Debugging	Locating and fixing errors, verification of functions	Anthropic Claude
Writing		
Language	Correction of spelling, grammar and wording	Grammarly
Formatting	Templates for LaTeX figures and tables	Anthropic Claude
Without AI support		
Research design	Research question, choice of models and data, design of the experiments	–
Experiments	Execution of all training and evaluation runs	–
Conclusions	Scientific assessment of the results and the conclusions drawn from them	–

Declaration of Authorship

This thesis is a presentation of my original research work. Wherever contributions of others are involved, every effort is made to indicate this clearly, with due reference to the literature and acknowledgement of collaborative research and discussions. The use of AI-based tools is documented in the preceding section.

Munich, 30.09.2026 C. Vatt

Place, Date, Signature