TECHNISCHE UNIVERSITÄT WIEN

Evaluating Sentinel-2 Super-Resolution Algorithms for Automated Building Delineation

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Geodäsie und Geoinformation

eingereicht von

Samuel Hollendonner, BSc
Matrikelnummer 11826122

an der Fakultät für Mathematik and Geoinformation

der Technischen Universität Wien

Betreuung: Univ.Prof. Dr.rer.nat. Wouter Dorigo, MSc

Mitwirkung: Prof. Dr. Luis Gómez-Chova

Wien, 17. Oktober 2025

Samuel Hollendonner	Wouter Dorigo

TECHNISCHE UNIVERSITÄT WIEN

Evaluating Sentinel-2 Super-Resolution Algorithms for Automated Building Delineation

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Geodesy and Geoinformation

by

Samuel Hollendonner, BSc
Registration Number 11826122

to the Faculty Mathematics and Geoinformation

at the TU Wien

Advisor: Univ.Prof. Dr.rer.nat. Wouter Dorigo, MSc

Assistance: Prof. Dr. Luis Gómez-Chova

Vienna, 17th October, 2025

Samuel Hollendonner | Wouter Dorigo

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

Samuel Hollendonner, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 17. Oktober 2025

Samuel Hollendonner

Acknowledgements

I want to express my sincere gratitude to everyone who guided and supported me throughout the work on this thesis. First, to my colleagues at the IPL in Valencia, who not only welcomed me into their research institute but also shared valuable insights, methodologies, and design approaches that greatly shaped both the theoretical and practical framework of this research. Second, I am deeply thankful to my current and former supervisors at the Technical University of Vienna, whose mentorship introduced me to the foundations of scientific work and emphasised the importance of reproducible and FAIR research. Finally, I extend my heartfelt thanks to my family and friends for their support and encouragement. Without the contributions of all those mentioned, this work would have been far more difficult - and far less enjoyable.

Abstract

Super-Resolution is the result of enhancing the spatial resolution of images while introducing high-resolution details and has been proposed as a way to make satellite data such as Sentinel-2 more useful for real-world applications. It promises higher-resolution imagery at a low cost, as existing image sources can be relied on. While many Super-Resolution models with varying architectures and training datasets have been developed, independent frameworks to evaluate their effectiveness in real-world tasks remain limited.

This thesis examines the performance of Super-Resolution for Sentinel-2 imagery on the downstream task of building delineation using a novel high-quality dataset covering Austria, created specifically for the proposed task. The methodology consists of three main steps: first, constructing a spatially and temporally aligned dataset with orthophotos as high-resolution references, Sentinel-2 images as lower-resolution inputs, and cadastral masks as ground-truth labels; second, applying Super-Resolution models to super-resolve Sentinel-2 images from a spatial resolution of 10 m to 2.5 m. In parallel, the Sentinel-2 images are upsampled to 2.5 m using interpolation methods to provide a deterministic baseline for evaluating the Super-Resolution results; and third, training UNet models for building delineation on both super-resolved outputs and interpolated Sentinel-2 images.

Three main findings emerge: first, orthophoto-based building delineation achieves the best results; second, models trained on interpolated images outperform those using Super-Resolution outputs; third, the differences between Super-Resolution models demonstrate that their choice of training data and architecture has a considerable influence on performance. These results suggest that, for building delineation, Super-Resolution currently offers no advantage over interpolation methods, and further development is needed to justify its real-world application. Furthermore, using original high-resolution image sources allows the most accurate result, if such data is available. More broadly, the outcomes highlight the importance of structured evaluation frameworks for benchmarking Super-Resolution on downstream tasks. Extending such evaluations to include diverse real-world applications will be essential for advancing Super-Resolution towards robust and task-oriented models capable of delivering reliable super-resolved satellite imagery.

Contents

CHAPTER 1

Introduction

This chapter provides the overall introduction to the thesis, outlining its motivation and objectives, and defining the Research Questions (RQs) which will guide the study. All code repositories and datasets supporting this work are publicly available under an open-source license: https://github.com/Zerhigh/Evaluating_Sentinel-2_Super-Resolution_Algorithms_for_Automated_Building_Delineation. A digital version of this thesis with colourised figures is also be available in this repository.

1.1 Problem Statement and Research Questions

Satellite earth observation plays a central role in monitoring, managing, and understanding environmental and urban processes. Freely available datasets provided by national and international agencies such as ESA and NASA offer global coverage and high temporal resolution, but are generally limited in spatial resolution (e.g., 10 m for Sentinel-2's Red, Green, and Blue (RGB) and Near-Infrared (NIR) bands and 30 m for Landsat-8). In contrast, private companies provide Very High Resolution (VHR) imagery at spatial resolutions down to 0.3 m with, e.g., WorldView-3.

This gap is particularly relevant given the inequalities in access to VHR satellite imagery. Institutions in high-income countries can purchase commercial data or generate orthophotos through aerial campaigns themselves [1], whereas those in low- and middle-income countries often depend on freely available but lower-resolution images such as from Sentinel-2. If Super-Resolution (SR) algorithms can effectively enhance the spatial resolution of such imagery, they could help mitigate disparities in geospatial data access and expand the usability of open data sources to new domains. Moreover, SR techniques could provide particular value in disaster situations, where free lower-resolution imagery is typically available more rapidly than commercial VHR data enabling faster and more detailed situational assessments. Additionally, the trustworthiness of SR algorithms is especially important in such scenarios, as the distribution of relief or rescue operations

require not only High Resolution (HR) data, but also robust and reliable super-resolved images.

Although Deep Learning (DL)-based SR is a relatively recent research field, a considerable number of algorithms have already been developed [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17]. Evaluating such SR algorithms is typically carried out through pixel-wise comparisons with reference images and dedicated SR-specific metrics on benchmarking datasets [18]. However, such approaches do not necessarily capture the practical utility of SR imagery in downstream applications. In response, more recently developed SR methods include an evaluation in real-world scenarios where the super-resolved imagery is applied to tasks such as water body detection [8], land cover classification [14], field delineation [17], or flood and methane plume detection [11]. Yet, these evaluations are often limited in scope, focusing on individual applications without systematic comparison across different SR approaches. The research field would therefore benefit from an evaluation framework that independently assesses the performance of SR methods to determine if they provide meaningful improvements.

To address these research gaps, this thesis investigates the applicability of state-of-the-art SR algorithms to a representative real-world task: building delineation in Austria. The study focuses on assessing whether the application of different SR models to Sentinel-2 imagery can improve model performance relative to interpolated Sentinel-2 data and how these results vary across different settlement types. Based on these objectives, the following RQs were formulated:

1. Are SR algorithms advanced enough for their super-resolved output to be used in real-world applications?
2. Do certain SR algorithms perform better than others, and what impacts these differences?
3. Do SR algorithms perform better in urban, semi-urban, or rural areas?
4. Can SR algorithms adapt to the unique spectral and spatial image composition of Austrian imagery, and can they be applied globally?

1.2 Research Approach

To answer these RQs, a systematic evaluation of multiple SR models applied to Sentinel-2 imagery is conducted. The super-resolved outputs are used to train DL models for building delineation, with models trained on interpolated Sentinel-2 imagery serving as a deterministic baseline. An additional model trained on HR orthophotos is included to validate the selected methodology.

The experimental setup covers diverse settlement types across Austria, including unpopulated regions, rural villages, suburban zones, and dense urban centres, to enable a robust

assessment of algorithm performance in heterogeneous environments. The analysis draws on three principal datasets: first, cadastral building footprints accessed by Bundesamt für Eich- und Vermessungswesen (BEV); second, VHR orthophoto imagery with a native spatial resolution of 0.2 m resampled to 2.5 m; and third, temporally and spatially matched Sentinel-2 imagery obtained from Google Earth Engine (GEE) which will be super-resolved to 2.5 m. For all data sources, the RGB and NIR spectral bands are used due to their availability and their expected benefit for the task of building delineation.

For clarity throughout this thesis, spatial resolutions are categorized as follows: VHR refers to imagery up to 2.5 m, typically derived from aerial or commercial sources; HR denotes the resolution range of 2.5–5 m corresponding to super-resolved and resampled outputs; and Low Resolution (LR) applies to native Sentinel-2 imagery at 10 m and above.

1.3 Thesis Structure

The remainder of this thesis is organized as follows. Chapter 2 reviews relevant literature on SR and building delineation. Chapter 3 introduces the creation of datasets, outlines the respective preprocessing workflows, and details the experimental setup. Chapter 4 presents samples of super-resolved images and the results obtained from the building delineation experiments, followed by the interpretation of results and answering of the RQs in Chapter 5. To conclude, Chapter 6 summarizes the key findings and provides an outlook for future research.

CHAPTER 2

Literature Review

This chapter reviews related work that forms the foundation for assessing SR performance in downstream applications. It summarizes relevant developments in remote sensing-based SR research, as well as approaches to DL-based building delineation that inform the methodological design presented in Chapter 3.

2.1 Interpolation Methods

While complex SR algorithms are being developed, interpolation techniques have been used to upsample images for decades [19]. As interpolation methods do not introduce any high-frequency information in the resulting image [20], they can not be considered SR algorithms themselves, but are capable of smoothing and enhancing signal input. Thus, they enable advanced operations for feature extraction from interpolated images [21]. Furthermore, interpolations are widely used as structural components (layers) in DL-based SR models and can act as a baseline technique for comparing newly developed SR algorithms [22, 23, 24]. Various interpolation techniques exist, with unique kernel shapes and sizes, functions, and implementations across different programming environments. As several different interpolation methods will be used during the processing for this thesis, a selection, including advantages and disadvantages, will be explored here.

Nearest Neighbor (NN) interpolation is the simplest interpolation method, assigning the value of the closest neighbouring pixel to each interpolated point. While it is computationally efficient, it produces blocky features that are undesirable in most contexts [25]. This interpolation method generates a resampled image, which is spectrally and spatially identical to its input, albeit with a higher sampling rate. NN is the preferred interpolation method for resampling classification maps (especially binary ones), as they require a consistent class assignment, which could be interfered with when using other interpolation methods.

Bilinear interpolation is characterised by aggregating the weighted average over the four closest pixels using a linear interpolation. It is the interpolation of choice for applications that rely on fast and reliable results, as it smooths and blurs the input without introducing any artefacts [25].

Bicubic interpolation extends bilinear interpolation by using a larger kernel (16 pixels instead of four) and applying a third-order polynomial function instead of a linear one. This technique can preserve fine details better, but may introduce artefacts such as ringing [21].

Despite the theoretical definition of these algorithms, their implementations vary significantly across programming languages and packages. This is particularly important in Machine Learning (ML) workflows, which often rely on the repeated resampling of input data. Venturelli [26] conducted experiments with several well-known Python libraries and concluded that *Pillow* [27] is the most reliable library available, as it implements anti-aliasing filters automatically. Thus, any image interpolation and resizing applied for experiments conducted within this thesis are achieved by using *Pillow* functions inside *PyTorch* wrappers.

2.2 Super-Resolution

SR refers to a range of techniques aimed at enhancing the spatial resolution of an image by reconstructing HR features, also referred to as high-frequency information, from one or more LR observations. This introduction of high-frequency information can be achieved with various types of models, either image processing or DL-based ones, which increase the spatial resolution of the LR input image and simultaneously try to predict HR features in the output. SR is an inherently ill-posed problem because the mapping from LR to HR can yield an infinite number of possible solutions [28]. The performance of SR varies significantly depending on the chosen model architecture, loss function, and dataset, all of which define the system's capabilities for specific applications. This section outlines the fundamental principles of SR, its application on remote sensing data, and presents a selection of SR model in detail.

2.2.1 Super-Resolution Overview

SR methods have been developed for decades; early approaches were grounded in traditional image processing, whereas modern approaches rely on DL. Regardless of the algorithm behind it, SR can be categorised into either multi-image or single-image approaches, depending on whether a sequence of images or individual ones are used.

2.2.1.1 General Development of Super-Resolution

Traditional SR methods are typically based on a forward degradation model that must be inverted to recover HR images. Farsiu et al. [29]'s approach involved the fusion of an image sequence with sub-pixel shifts between subsequent frames, which allowed the introduction

of HR frequencies as information could be extracted and merged from overlapping, but differently sampled, LR images. Glasner, Bagon, and Irani [30] introduced a framework leveraging the recurrence of small image patches across different scales to perform example-based single-image SR without needing external training data. These methods usually minimise the L2 norm between the observed LR inputs and the HR outputs estimates from the forward model.

With the introduction of DL models and their increasing capabilities, novel SR methods have been developed, which can be used on more general inputs while introducing new issues and considerations. Dong et al. [31] introduced Super-Resolution Convolutional Neural Network (SRCNN), a Convolutional Neural Network (CNN) trained on a natural image dataset and Mean Squared Error (MSE) as a loss function to optimize the Peak Signal-to-Noise Ratio (PSNR). While PSNR was once considered a reliable proxy for perceptual quality, its limitations have been assessed, and newer perceptual metrics have been proposed that better reflect the perceived quality of super-resolved images [32]. The invention of Generative Adversarial Networks (GAN) models defined another substantial leap in SR, as generative models aim to reconstruct the natural attributes of super-resolved images. Wang et al. [33] published the model Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN), which focuses on introducing high-frequency information while retaining and improving the human-perceived perception. This is achieved by creating a novel MINC loss, an extension of the Perceptual Loss defined by Johnson, Alahi, and Fei-Fei [34], which uses the activation layers of a pre-trained network to maximise the similarity between recognised features. More recently, Latent Diffusion Models have been modified to enable SR by transforming images into a latent space using an autoencoder and then applying a denoising diffusion process [35]. Although this method can generate high-quality outputs with rich semantic detail, its computational complexity results in long training and inference times. To evaluate perceptual similarity, the Learned Perceptual Image Patch Similarity (LPIPS) metric [36], which compares deep features extracted with a pre-trained CNN, was employed, offering improved alignment with human visual perception, albeit at a high computational cost.

In summary, SR methods have evolved from classical image processing algorithms to DL-based ones, with basic CNNs, generative, and diffusion models being the ones most widely used. With this transition, the issue of models creating or omitting faulty high-frequency information becomes more urgent and problematic, as models' reasoning can not be retraced, and validation becomes difficult. The introduction of such artefacts can be especially problematic in research fields such as remote sensing, where output fidelity matters more than visual quality [37]. In light of the recent development of SR models, arguments have been made about whether to develop SR to be visually pleasing to human observers, and risk including hallucinations and artefacts, or if SR should facilitate machine-consumption for downstream tasks [2].

2.2.1.2 Super-Resolution in the Context of Remote Sensing

Applying SR to remotely sensed images introduces new challenges, while allowing for an increase in the spatial resolution of input images. SR methods can introduce high-frequency information, which is especially useful for downstream tasks, as it can allow the detection of small objects which were not detectable in the LR image before. This impact has been observed by Lac et al. [8] and Ayala et al. [38] who underlined this importance for scenarios in which demand for HR image data can not be met. However, applying SR to remote sensing introduces new challenges. Satellite or aerial imagery is captured from long distances, covers diverse and often complex landscapes, and undergoes heavy processing algorithms (e.g., orthorectification) to retrieve images from raw sensor output. As a result, models developed for natural images taken with handheld cameras do not directly generalise to remote sensing data and must be retrained on domain-specific datasets [39].

One major challenge is the limited availability of suitable HR training data for remote sensing. Two common options for generating datasets as foundations for SR models, synthetic and cross-sensor, have been developed to address this. Synthetic datasets are created by degrading HR images using downsampling, noise, blur, and other morphological operations [40]. However, this can limit generalisation capabilities, as models may only learn to reverse the synthetic degradation rather than generalise to real-world LR images due to overlooked biases between the synthetic and real LR distributions [37]. Recent approaches produce HR-LR pairs by training DL models to apply harmonisation, noise, and blurring to HR images [41]. To the best of my knowledge, no independent study has evaluated whether using DL to generate synthetic datasets helps SR models generalise, rather than simply learning to reverse the applied degradation as well. Cross-sensor datasets combine images from different imaging platforms, such as pairing Sentinel-2 LR with HR imagery from satellite or aerial-based sensors. Acquiring these data sources can be difficult, as HR data is often commercial or only available with specific spatial and temporal constraints. This can create issues such as spatial, temporal, and spectral misalignment due to differences in capture timing, sensor characteristics, and processing algorithms. With careful pre-processing, such as image co-registration and band harmonisation, these challenges can be mitigated to a certain degree [3, 42, 43].

As creating SR datasets is labour-intensive and often subject to licensing restrictions, only a few open-source datasets are publicly available, and these are usually very task-specific. As a result, most SR implementations are first tasked with creating custom datasets to fit their requirements, before training a model is viable. Furthermore, the choice of dataset has a profound effect on a model's generalizability, especially as it is a tremendous challenge to encompass all of Earth's diverse landscapes in a single dataset. Additionally, the model's architecture and selected training dataset impact the risk of generating hallucinated structures that appear visually plausible but have no correlation to real-world features. Such artefacts can undermine downstream applications such as building delineation, where geometric accuracy is critical.

Several datasets have been developed to support training and evaluation of SR models on remote sensing imagery, each with different design goals, image sources, and approaches for generating HR-LR image pairs. The following selection does not aim to be a complete aggregation of available datasets, but a concise summary to provide an overview of different data sources and processing algorithms. They are presented in chronological publication order:

- **SEN2VENμS** is a single-image cross-sensor dataset that pairs Sentinel-2 as the LR source with VENμS as the HR reference. VENμS provides observations at a spatial resolution of 5 m across 12 spectral bands, with its RGB bands closely matching the corresponding Sentinel-2 bands. The dataset includes all 10 m and 20 m Sentinel-2 bands together with the radiometrically adjusted VENμS bands, provided in the Level-2A (L2A) format. Cloud-free samples are selected from acquisitions on the same day and cover diverse landscapes worldwide, ensuring high variability in scene types [44].

- **Worldstrat** is a multi-image cross-sensor dataset with matched Sentinel-2 LR and SPOT6/7 HR imagery. The HR images include a panchromatic band with 1.5 m as well as RGB and NIR bands with 6 m spatial resolution. The dataset is sampled globally, with a particular emphasis on human settlements, non-settled areas, as well as areas usually under-represented in similar datasets, such as small-scale mining and displaced populations. HR-LR pairs are temporally aligned and filtered for cloud cover [42].

- **MuS2** is a multi-image cross-sensor dataset combining two sources: Sentinel-2 LR and WorldView-2 HR images, with a panchromatic and eight other bands ranging from visible to NIR, resampled to a spatial resolution of 3.3 m. The dataset is intended as an evaluation dataset for SR, with images sampled to focus on European cities. A large temporal offset of up to eleven years between the two sources was compensated for by excluding areas which feature dramatic differences by applying a change mask. This, together with a cloud filtering algorithm, should guarantee a high degree of similarity between HR and LR data [3].

- **SEN2NAIP** includes two single-image subsets, first a cross-sensor and second a synthetic dataset. Both use Sentinel-2 LR and National Agriculture Imagery Program (NAIP) HR images and focus on retaining the spectral properties of both sources by harmonising them with each other. The cross-sensor dataset relies on strict temporal alignment (within one day) to ensure compatibility with Sentinel-2 imagery, and is used to train degradation models for generating LR image samples from NAIP HR to create the synthetic dataset [41]. Enlarged versions of this dataset exist, which are not published at the moment of writing this thesis.

2.2.2 Deep Learning based Super-Resolution Implementations for Remote Sensing

While many different DL-based SR algorithms have been developed, a small selection will be discussed in detail due to their relevance for subsequent experiments. This allows to understand their similarities and highlight differences in model architecture, training setup, and used datasets. The following SR models are presented in chronological order of publication, and a tabular overview summarising their distinct features and model architectures is provided in Table 2.1.

2.2.2.1 SR4RS

Cresson [13] developed the SR4RS model based on prior development of the Orfeo ToolBox [45] for spatial data processing and the single-image GAN-based SR approach implemented by Ledig et al. [46]. The GAN architecture was extended by including the NIR band in addition to the already implemented RGB-SR. SR4RS is trained on a cross-sensor dataset covering mainland France with Spot-6 and Spot-7 images as HR, and temporally matched Sentinel-2 images as LR images. HR images were resampled to 2.5 m and radiometrically calibrated to match Sentinel-2's spectral characteristics. The pre-trained model was published in an open source repository, and its use for inference was streamlined in the *superIX* framework.

2.2.2.2 Evoland

The Evoland model was developed by Lac et al. [8] to super-resolve all 10 m and 20 m Sentinel-2 bands to a spatial resolution of 5 m. In their approach, the 20 m bands are bicubically upsampled and concatenated with the 10 m bands before being super-resolved with a Real-ESRGAN architecture. Training was conducted using the SEN2VENμS dataset [44], which provides a global collection of temporally and radiometrically matched image pairs from Sentinel-2 and VENμS. The model is accessible for inference with the *superIX* framework.

The authors evaluated their model in two ways. First, super-resolved Sentinel-2 images were compared against VENμS reference images using PSNR. Results showed that the SR model performs on a comparable level to bicubic resampling for the 10 m bands, while yielding improvements for the 20 m bands. Second, performance was assessed on a downstream task: water body detection in an agricultural region in France. Using a spectral-index-based algorithm, the authors demonstrated that super-resolved Sentinel-2 imagery, particularly the enhanced 20 m bands, improved the identification of small water bodies that were not detectable at the native resolution.

2.2.2.3 DeepSent

Tarasiewicz et al. [4] use spectral and temporal information fusion for super-resolving all Sentinel-2 bands to a common resolution of 3.3 m. First, available images are fused in the temporal domain, followed by a fusion in the spectral domain by grouping bands of similar resolutions into subsets and upsampling them stepwise: from 60 m to 20 m, then from 20 m to 10 m, and finally all bands in unison to 3.3 m. Their CNN-based model is trained on a variety of datasets. A synthetic Sentinel-2 based dataset was generated by applying several filters, including sub-pixel shifts, contrast and brightness manipulation, and Gaussian blurring to selected Sentinel-2 tiles, which were bicubically downsampled to represent the 3.3 m target resolution. Similarly, a second synthetic dataset was created by inverting this process with 3.4 m HR multispectral imagery from Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) and upsampling it to the corresponding Sentinel-2 band resolutions of 10 m, 20 m, and 60 m. A third dataset was created by acquiring nine temporally differing Sentinel-2 image patches for each location used in the first dataset. No augmentations were applied to these tiles, and they were used for testing their model's output without a HR counterpart. The fourth dataset, MuS2 (see Section 2.2.1.2) [3] was also used for testing purposes.

The evaluation of their trained model was achieved by comparing it against prior implementations such as RAMS [47], HighRes-net [37], and DSen2 [6] with the metrics PSNR and Structural Similarity Index Measure (SSIM), modified to compensate for brightness and pixel displacements between HR and LR image pairs. According to their experiments, the presented model, DeepSent, outperforms these other models, as the unique implementation of temporal and spectral fusion improves the metrics results and reduces artefacts.

2.2.2.4 Swin2Mose

Rossi et al. [14] introduced the Swin2Mose model architecture, extending the Shifted Window Transformer (SWIN)-based SR framework Swin2SR by incorporating an additional MoE-SM layer. The model is designed to super-resolve 10 m and 20 m Sentinel-2 bands to a spatial resolution of 5 m. Training and evaluation were conducted using two datasets: SEN2VENμS and OLI2MSI, a cross-sensor dataset merging Landsat OLI as LR with Sentinel-2 MultiSpectral Instrument (MSI) as HR, [48]. The use of these diverse datasets enables performance assessments at the different scales of 2×, 3×, and 4× spatial resolution improvements.

While the primary objective was to validate the newly proposed network design, the authors also evaluated downstream performance on land cover classification using the SeasoNet dataset [49]. Results demonstrate that Swin2Mose outperforms comparable Swin-based SR models and improves segmentation accuracy relative to the SeasoNet baseline. The model and pre-trained weights are publicly available on GitHub, and the implementation is also integrated into the *superIX* framework.

2.2.2.5 LDSR-S2

The LDSR-S2 model published by [10] is an adaptation of Latent Diffusion [35] to the realm of remote sensing, specifically to the task of super-resolving Sentinel-2 bands to 2.5 m. By modifying the network architecture and regularising the input images in the latent space, the model is capable of better retaining original image structures with a particular focus on the spectral consistency of LR input images and super-resolved outputs. Furthermore, the probabilistic characteristic of diffusion models allows the construction of pixel-level uncertainty maps, which enables users to directly evaluate the reliability and trustworthiness of model outputs.

The model is trained on the SEN2NAIP dataset [41] and was compared with other SR models such as Satlas SRGAN [2] or SR4RS [13], on the spectral, radiometric, and pixel-based metrics proposed in the OpenSR-Test benchmarking framework [18]. LDSR-S2 outperformed the compared models in all metrics but Synthesis, which describes the preservation of features from the super-resolved compared to the LR image sample.

2.2.2.6 SEN2SR

Aybar et al. [11] conducted experiments with various DL based SR architectures to develop a model capable of upsampling all 10 m and 20 m bands of Sentinel-2 to a uniform spatial resolution of 2.5 m. This encompasses the RGB, NIR, and Short Wave Infra-Red (SWIR) bands and allows the seamless integration of algorithms dependent on a consistent resolution and spatial alignment across all bands. Apart from increasing the spatial resolution, the authors focus on maintaining the spectral consistency output of their models, which is essential for applications relying on reflectance estimations, but less important for task-driven approaches such as feature extraction. This was achieved by including a low-frequency hard constraint in the model.

The authors of SEN2SR conducted experiments with several datasets: S2NAIPv2, an updated version of the S2NAIP dataset [41]; the cloud-free images from CloudSEN12 [50, 51, 52] for training the SWIR upsampling; and OpenSR-Test [18]. Three model types were tried out: CNN, SWIN, and Mamba. To further evaluate the performance of the provided models, the authors tested them on two distinct downstream tasks, which should benefit from access to super-resolved imagery: flood (WorldFloods [53]) and methane plume (MARS-S2L [54]) detection. The authors concluded that SR models with a large number of tunable parameters (>15 million) did not achieve significantly better results than smaller ones, and that the Mamba-based model outperformed the CNN version [41].

2.2.2.7 Tracasa

Tracasa is a private company developing SR algorithms for various applications such as small object detection [38], Sentinel-1 SR [55, 56], and road and building segmentation [57, 15]. While research results are published in scientific journals and at conferences, detailed model descriptions and training setup parameters are not provided. Their most recent SR model was provided after contact was established, and the authors provided

the following information. The model uses the Second-order Attention Network (SAN) for single-image SR [58], and super-resolves the RGB and NIR bands of Sentinel-2 to the spatial resolution of 2.5 m. No information is provided regarding the training dataset or downstream evaluation tasks. Implementation details for this proprietary model are provided in Section 3.3.3.1.

Name	Model Architectures	SR-bands	Resolving Resolution [m]	Train Dataset	Test Dataset	Downstream Task Dataset	Single- or Multi-Image	Publicly Available
SR4RS	GAN	B02, B03, B04, and B08	2.5	Spot-6 and Spot-7 Sentinel-2	Spot-6 and Spot-7 Sentinel-2		Single-Image	Code & Weights and in *superIX*
Evoland	Real-ESRGAN	B02–B12	5	SEN2VENμS [44]	SEN2VENμS [44]	Water body detection	Single-Image	In *superIX*
DeepSent	CNN	B01–B12	3.3	synthetic Sentinel-2 synthetic AVIRIS	Sentinel-2 timeseries MuS2 [3]		Multi-Image	Code & Weights upon request
Swin2Mose	Swin2SR with MoE-SM layer	B02–B12	5	SEN2VENμS [44] OLI2MSI [48]	SEN2VENμS [44] OLI2MSI [48]	SeasoNet [49]	Single-Image	Code & Weights and in *superIX*
LDSR-S2	Latent Diffusion	B02, B03, B04, and B08	2.5	Customized OpenImage [59] SEN2NAIP [41]	Opensr-Test [18] SEN2NAIP [41]		Single-Image	Code & Weights
SEN2SR	CNN, Mamba, SWIN	B02–B12	2.5	synthetic S2NAIPv2 CloudSEN12 [50, 51, 52]	Opensr-Test [18] cross-sensor SEN2NAIPv2	WorldFloods [53] MARS-S2L [54]	Single-Image	Code & Weights
Tracasa	SAN	B02, B03, B04, and B08	2.5	Unknown	Unknown	Unknown	Single-Image	No, inference upon request

Table 2.1: Comparison of SR models and their training parameters.

2.2.3 Benchmarking of Super-Resolution Algorithms

With many different SR models being created by research groups and commercial companies alike, independently developed methods for evaluating and comparing them against each other are rare. This is partially due to the complexity of the underlying SR algorithms, which limits the interoperability and comparability of different implementations. An additional factor is the difficulty in devising metrics which evaluate aspects not considered as loss functions during the training of SR algorithms. This section comprises the various issues which arise during the evaluation of SR algorithms, and provides examples for SR specific benchmarks with their advantages and disadvantages.

While benchmarks incorporate tailored datasets and specifically created metrics, easier methods for comparing SR exist in the form of standardised metrics. Using such metrics, e.g. pixel-based ones like PSNR, MSE, or SSIM, or the AI-based LPIPS [32], allows one to understand the performance of each model on its test dataset and does not require additional sophisticated methodology. Furthermore, if models are published under open-source licenses, researchers can access and apply them to novel datasets to evaluate and compare different SR implementations.

Relying solely on metrics to evaluate model performance is sufficient for training and adapting one's model during development, but, if applicable, models should be evaluated on a domain-specific benchmark to assess and fully understand their performance [60]. Available benchmarks suited to Sentinel-2 SR include MuS2 (see Section 2.2.1.2) [3], B4MultiSR, which focuses on multi-image SR with images from Sentinel-2 and further sensors of spatial resolutions ranging from 0.3-5 m [61], and the recently published OpenSR-test benchmark [18]. OpenSR-test is a collection of several small-scale datasets covering diverse landscapes and HR sensors, such as NAIP, SPOT, VENUμS, and aerial imagery from Spain. Additionally, it features three new metrics which go beyond

conventional metrics to provide a better understanding of SR capabilities: **Improvement**, describes the alignment of the super-resolved image with a HR reference image; **Omission**, includes all high-frequency information which the SR model was not able to capture; and **Hallucination**, is the introduction of high-frequency features which were not present in the HR reference. This benchmark has already gained traction and is used by several single-image SR models trained on Sentinel-2 imagery [5, 12, 10, 11]. Nevertheless, no benchmark should be regarded as a definitive measure of model performance. While in some DL domains benchmark improvements can be definitive of meaningful progress, this assumption does not always hold in remote sensing, where minor gains on benchmark datasets may not translate into significant benefits in real-world applications. Data variability, sensor noise, and domain-specific constraints often limit the generalizability of models [62].

Applying developed SR models on downstream tasks allows their evaluation in regard to specific applications. In contrast to the previously mentioned benchmarks, downstream tasks do not apply metrics directly to the image output of each model, but require an additional processing layer. This could include spectral-index-based classification or feature extraction algorithms, which are applied to the super-resolved inference images from a model. While several SR models are already evaluated on such defined use-cases [11, 2, 12, 63], there is a lack of externally defined and created downstream tasks which can be used to universally compare SR models and evaluate their impact on real-world applications.

2.3 Building Delineation

Detecting buildings from remote sensing imagery is essential for monitoring urban growth, supporting disaster response, and assessing vulnerable environments. The task is challenging due to the limited availability of high-quality globally distributed imagery and reliable Ground Truth (GT) annotations. Additional difficulties arise from the diversity of building shapes, building materials, and sizes across urban and rural settings, as well as from data-specific issues such as off-nadir effects on tall structures [64].

Most delineation approaches rely on public sources that provide imagery [65, 64, 66, 67, 68, 69] and GT building labels [70, 71]. These sources are typically limited to VHR images below 1 m spatial resolution, where footprint extraction performs well and performance improvements are achievable. In contrast, fewer datasets and models exist for coarser resolutions, where building edges blur and extraction becomes more difficult. Consequently, research has largely relied on VHR satellite, aerial, or Unmanned Aerial Vehicle (UAV) imagery, which are costly or available only at low temporal frequency. Super-resolved Sentinel-2 imagery could provide an accessible, globally available alternative.

2.3.1 Deep Learning Approaches for Building Delineation

Building delineation is a challenging task due to the diversity of shapes, construction materials, and cultural styles of buildings, which complicates labelling. Traditional (non-DL) methods rely on human annotation, semi-automatic tools, or spectral indices [72, 73, 74]to detect building footprints, but these approaches are slow and can fail to capture the full range of buildings.

Recent DL-based methods address these challenges by enabling automated extraction of building footprints at a large scale. Accuracy depends on image quality, algorithm design, and GT quality, but DL approaches can map large areas in short time spans. However, because most implementations rely on similar spatially limited VHR datasets [66, 67, 68, 69], their generalisation capacity remains limited. Most implementations define delineation as a segmentation task and evaluate model performance using Intersection over Union (IoU), F_1, and additional raster- or vector-based metrics.

Several datasets have supported model development. For example, Wei et al. [75] introduced a CNN-based model that extracts already polygonised footprints from VHR aerial imagery (WHU [67], WHU-mix [68]) and the vectorised World Building dataset [70]. Li et al. [76] applied a region-based CNN with a ResNet-34 encoder to the WHU dataset with manually labeled GT. Li et al. [77] proposed the CNN-based HD-Net, trained on combined datasets such as Massachusetts [69], WHU, and Inria [78], achieving competitive results. These examples highlight the applicability of CNN-based models for the task of building segmentation.

Benchmark initiatives, particularly the SpaceNet challenges (1, 2, 4, 7, and 8), have further advanced the field of building delineation. Challenges 1 and 2 used VHR imagery (30-50 cm) from cities such as Las Vegas, Paris, Khartoum, Rio de Janeiro, and Shanghai, with top solutions using UNet or Random Forest models [65]. Challenge 4 focused on off-nadir imagery, where UNet variants with extensive post-processing improved results over their competitors [64]. Challenge 7 shifted to coarse 4 m resolution time-series data, with High-Resolution Net (HRNet) [79] proving effective at retaining spatial detail at such a resolution [80]. Similarly, Challenge 8 provided HR satellite imagery and targeted multi-class segmentation of roads and buildings before and after flooding events. This challenge was also won with an HRNet implementation, which highlights its capabilities for difficult segmentation tasks [66].

Beyond CNNs, newer model architectures have been tested for building delineation. Osco et al. [81] repurposed the Segment Anything Model (SAM) model [82] for multi-class segmentation across VHR images from UAV, aerial, and satellite sources. While it achieved good results, post-processing was required to isolate buildings from the other segmented classes. Ayala et al. [83] explored diffusion-based models on the Massachusetts dataset [69], achieving accuracy comparable to an HRNet but at significantly higher inference costs due to the iterative nature of diffusion models.

In summary, DL-based delineation methods show strong performance on VHR and HR imagery and make extended use of CNN architectures. GT data is typically sourced

from open datasets or manually annotated. While more advanced models are emerging, performance improvements are difficult to measure, as few global benchmarking datasets exist, which would make such a comparison possible. All implementations are limited by their dependence on VHR, region-specific data, which can be difficult to acquire and limits the models' generalisation capabilities on spatially differing study sites.

2.3.1.1 Model Architecture Overview

The previous analysis suggests that two major model architecture types were used for successful building delineation. UNets, with various backbones, and HRNet are widely deployed and provide reasonable results while consuming fewer resources than newer and more complex architectures.

UNet is one of the most widely used DL architectures. Since its introduction by Ronneberger, Fischer, and Brox [84] for semantic segmentation in medical images, it has been adapted for many tasks, including road extraction [65], building footprint extraction [64, 85, 86, 87], and more. Recently, UNet has also served as a baseline in benchmarking applications for foundation models in earth observation [88]. Its popularity arises from its simple yet effective design and its adaptability across diverse application domains. Standard UNet architectures are available in almost all DL frameworks, which allows an easy implementation into workflows.

The UNet architecture follows a symmetric U-shape: an encoder downsamples the input to capture contextual features, while a decoder upsamples to reconstruct pixel-wise predictions at the input resolution. Skip connections help preserve spatial details lost during downsampling. A key advantage is its flexibility, as UNet can be combined with various backbones. ResNet [89] and its follow-up improvement ResNext [90] are common backbones used for semantic segmentation tasks, and are available in different configurations. These include the number of layers, which enables models to solve more complex tasks but increases memory usage, or the inclusion of special modules such as Squeeze-and-Excitation (SE) blocks [91]. They allow the model to retain information between channels and emphasise the most relevant information. This wide selection allows users to create both lightweight and complex variants tailored to task-specific requirements. However, due to repeated down- and upsampling operations, UNet can struggle to retain HR information.

HRNet was originally developed for human pose estimation, semantic segmentation, and object detection, with a focus on retaining HR representations through multi-branch convolutional processing [79]. Unlike encoder-decoder architectures, HRNet maintains a HR stream throughout the network while also extracting semantic context at progressively lower resolutions. An Object-Contextual Representations (OCR) module can be added to improve segmentation quality, particularly at class boundaries, which is especially relevant for building delineation. These strengths make HRNet particularly well-suited for tasks requiring fine-grained segmentation, but at the cost of increased computational demand.

Contrary to UNet, there is no readily available network architecture available in DL frameworks. To use a HRNet model, it can be deployed as a backbone in other models, or as a stand-alone model through the authors open source publication. While the prior allows an easy implementation, albeit with limited functionality, the latter provides full customizability but requires expert knowledge to set up and configure correctly.

2.3.2 Building Delineation in Combination with Super-Resolution

With the rise of DL-based SR, a growing number of studies now incorporate or evaluate SR as a means to enhance building delineation performance. Approaches vary; some use an end-to-end model to jointly super-resolve and segment buildings, while others apply SR as a pre-processing step to improve spatial resolution before segmentation. These differences are compounded by the variability in datasets used, which differ in spatial resolution, coverage, and geographic diversity.

2.3.2.1 Joint end-to-end Models

Xu et al. [85] introduced ESPC_NASUNet, a model designed to perform both SR and building delineation in one step. Trained on resampled VHR aerial imagery and applied to Sentinel-2 data over Beijing, qualitative evaluation indicated improved segmentation quality, although the quantitative impact of SR remained ambiguous. Building on [85]'s work, Feng et al. [92], Zhang et al. [22], and Liu et al. [93] developed similar one-step frameworks using Sentinel-2 imagery and GT from Tiandi Maps. While Feng et al. [92] and Liu et al. [93] emphasised the generation of national-scale building maps, Zhang et al. [22] compared performance with and without SR. Although raster-based metrics showed little difference, qualitative inspection revealed better delineation of building edges in SR-enhanced inputs.

Ayala et al. [15] used merged Sentinel-1 and Sentinel-2 data to produce 2.5 m resolution building and road maps, learning to super-resolve the output mask directly by rasterising vector-based GT data to the output resolution of 2.5 m instead of Sentinel-2 native 10 m. Despite not featuring an explicit SR module, their architecture achieved improvements over traditional baselines.

Sirko et al. [94] introduced a teacher-student framework where the teacher is trained on HR imagery and the student is fed a time-series of 32 Sentinel-2 images. This method effectively bypasses explicit SR while still achieving HR-like output. Validation was performed using large amounts of human-labelled data across several continents, and allowed the conclusion that the proposed model is capable of extracting accurate HR labels without the need for large amounts of HR training data.

2.3.2.2 Super-Resolution and Segmentation

Nguyen et al. [86] applied Real-ESRGAN to super-resolve VHR aerial imagery and trained a DeepLabV3 model (with ResNet backbone) using manually labelled data from slum areas in India. The authors report noticeable improvements in delineating building edges as a result of the SR pre-processing. Guo et al. [95] assessed the effect of SR on datasets ranging from 0.075 m to 2.4 m resolution, using Efficient Sub-Pixel Convolutional Neural Network (ESPCN) for SR and UNet for segmentation. Their findings suggest SR offers benefits in some cases but not consistently across datasets.

Panangian and Bittner [96] used Sentinel-2 and NAIP imagery (via the SEN2NAIP dataset published by Wolters, Bastani, and Kembhavi [2]) to train an ESRGAN with localized embeddings. A SAM model was then trained on super-resolved tiles and Open Street Map (OSM) labels. While the approach outperformed the Satlas model, which struggled with hallucinations, it underperformed relative to a SAM trained on original NAIP imagery. The authors also emphasised that standard SR evaluation metrics like SSIM, PSNR, and LPIPS are insufficient for assessing downstream utility, further reinforcing the relevance of implementing building delineation as its own downstream evaluation task for SR models.

Illarionova et al. [97] trained their SR model on HR imagery from MapBox, xView [98], and the Massachusetts datasets. They used the downstream task of building delineation to evaluate their Sentinel-2 SR outputs with manually verified OSM GT data. Among the used models, DeepLabV3, SWIN, and Twins, SWINperformed best, though super-resolved Sentinel-2 still lagged behind original 2.5 m resolution imagery.

Debella-Gilo [87] used ESRGAN to super-resolve Sentinel-2 images to 2 m resolution, comparing them to both original 10 m Sentinel-2 and native 2 m imagery from various HR satellites, which was used as the HR reference for training the SR model. A UNet semantic segmentation model was trained to detect urban structures with data from the Norwegian mapping portal serving as the GT reference. While native HR data yielded the best results overall, vector-based IoU and object count evaluations highlighted the benefits of SR in identifying large buildings. However, detecting small buildings remained problematic, reducing performance across all metrics. Notably, raster-based metrics like F_1 and IoU showed negligible improvements when comparing original and super-resolved Sentinel-2 data.

In summary, while SR shows potential to enhance building delineation, its effectiveness varies across datasets and implementations. Most implementations assessing SR capabilities determined that building delineation works best on native HR images rather than super-resolved LR ones. A unified evaluation framework would be beneficial for objectively comparing performance across different pipelines and validating their capabilities of generalising to new spatial areas.

2.3.3 Low Resolution Segmentation Considerations

As discussed in Section 2.3.2, most building delineation methods segment at VHR ranges. This thesis addresses building delineation at the much coarser resolution of 2.5 m, which complicates the segmentation process considerably. Model design will depend on the methods applied to images of a similar resolution.

Approaches combining both SR and building delineation rely on HR images with around 2.5 to 5 m spatial resolution. Only a few approaches have addressed native HR segmentation. Among them, SpaceNet Challenge 7 [80] is noteworthy for working with 4 m resolution time-series data. Using such data requires architectural adaptations to handle coarse spatial detail and allows valuable insights for designing robust models. Consequently, the methodology developed in this thesis will draw inspiration for model design from these sources.

CHAPTER 3

Methodology

This chapter outlines the methodological framework developed to evaluate various SR implementations for the downstream task of building delineation. It introduces the data processing pipeline used to generate required datasets, the inference of SR models on Sentinel-2 imagery, and the training of building delineation networks on the resulting data. All orthophoto and Sentinel-2 figures display only the RGB bands.

3.1 Data Description and Used Software

The evaluation of SR implementations relies on multiple complementary sources to generate datasets and various software tools. Parts of this chapter, in particular Section 3.1.2 and 3.1.3, were already published as a short paper at the AGIT conference in July 2025 [99], and thus include similar content.

3.1.1 Sentinel-2 MSI

The Sentinel-2 satellite mission is part of the European Union's Copernicus Programme, designed for systematic Earth observation. The mission consists of two polar-orbiting satellites, Sentinel-2A (S2A) and Sentinel-2B (S2B), placed in the same sun-synchronous orbit but phased 180° apart. This configuration enables a global revisit time of approximately five days at the equator, with more frequent coverage at higher latitudes due to orbital overlap [100]. Each satellite is equipped with the MSI, which captures imagery in 13 spectral bands at varying spatial resolutions, with Table 3.1 presenting a detailed description.

Sentinel-2 provides several data product levels to support diverse remote sensing applications. Level-1C (L1C) contains Top-of-Atmosphere (TOA) reflectance values. This product is geometrically orthorectified and includes radiometric corrections but does not remove atmospheric effects like haze or scattering. L2A contains Bottom-of-Atmosphere

(BOA) reflectance, meaning it includes atmospheric correction and provides surface reflectance data more suitable for time-series analysis, vegetation monitoring, and classification tasks. The Sentinel-2 images processed to L2A will be used throughout this thesis, as they are most commonly employed by SR models. Both L1C and L2A products follow a standardised structure and are distributed in the SAFE format. Each product is projected in the corresponding Universal Transverse Mercator (UTM) coordinate system, depending on the latitude zone and hemisphere, and uses World Geodetic System 84 as the geodetic reference [101]. Saving the data this way ensures easy implementation into further workflows, but dramatically increases data storage requirements as the UTM-based system results in overlapping, and thus redundant, image tiles [102].

These datasets are hosted on multiple platforms for public access. The Copernicus Open Access Hub provides full-resolution downloads of all Sentinel-2 data products in one ecosystem. While this is advantageous for accessing and downloading full Sentinel-2 tiles, it is not possible to download small subtiles from this source directly, as such functionality is not provided for the SAFE data format. Furthermore, it is only allowed to issue a certain number of requests per month (10.000 as of September 2025 [103]). Thus, several options are available for accessing Sentinel-2 data in Cloud Optimized GeoTIFF (COG) formats: Amazon Web Services (AWS) hosts Sentinel-2 COGs in an S3 bucket, which can be queried and accessed via the AWS CLI or a Python script [104]. Sentinel Hub also allows COG download via an Application Programming Interface (API), but their service is limited in free data access via a credit system. GEE offers a cloud-based environment for analysis-ready data, which can be easily queried, filtered and downloaded [105]. Furthermore, GEE offers extensive precomputed metadata including cloud masks, and Python packages such as *cubexpress* allow a simplified parallelised download of large amounts of image data. As the use of GEE and *cubexpress* suited the required needs perfectly, they were used throughout this thesis.

3.1.2 Austrian Orthophoto

Publicly available satellite imagery, such as the Sentinel-2 and Landsat series, offers high temporal and spectral resolution but lacks the VHR required for fine-grained DL-based building delineation [106]. Other satellite sources, such as WorldView-3, PlanetScope, SPOT 6 & 7 or Pléiades, provide products with a higher spatial resolution but with limited access due to their high acquisition cost or limited availability. Aerial imagery, captured by planes or UAVs, offers high spatial resolution but comes with spatial and temporal limitations. Unlike satellite systems, aerial images cover smaller areas and cannot be acquired as frequently. However, when collected and provided by governmental agencies, this data is typically of high quality, as it must adhere to strict regulations and is usually made available in an open-access format [107].

Austria's digital orthophoto series provides open-access imagery with a 20 cm Ground Sampling Distance (GSD), collected independently by Austria's nine states at least every three years during the summer months. The aerial images are published as overlapping image tiles in one of three MGI/Austria Gauss-Krüger Coordinate Reference System

Table 3.1: Sentinel-2 bands with information of their central wavelengths (S2A and S2B) and spatial resolutions.

Band	**Description**	**Central Wavelength** [nm]	**Resolution** [m]
B01	Coastal aerosol	442.7 (S2A), 442.3 (S2B)	60
B02	Blue	492.4 (S2A), 492.1 (S2B)	10
B03	Green	559.8 (S2A), 559.0 (S2B)	10
B04	Red	664.6 (S2A), 665.0 (S2B)	10
B05	Vegetation red edge	704.1 (S2A), 703.8 (S2B)	20
B06	Vegetation red edge	740.5 (S2A), 739.1 (S2B)	20
B07	Vegetation red edge	782.8 (S2A), 779.7 (S2B)	20
B08	NIR	832.8 (S2A), 833.0 (S2B)	10
B8A	Narrow NIR	864.7 (S2A), 864.0 (S2B)	20
B09	Water vapour	945.1 (S2A), 943.2 (S2B)	60
B10	Cirrus detection	1373.5 (S2A), 1376.9 (S2B)	60
B11	SWIR	1613.7 (S2A), 1610.4 (S2B)	20
B12	SWIR	2202.4 (S2A), 2185.7 (S2B)	20

(CRS) zones: West (EPSG:31254), Central (EPSG:31255), and East (EPSG:31256) [108]. These tiles (see Figure 3.1a) follow the COG standard, allowing efficient range-request access with minimal local memory requirements [109]. BEV annually aggregates the most recent aerial images into orthophoto series to ensure continuos overlapping coverages, and publishes the series in their data catalogue. For this thesis, the latest orthophoto series published by BEV from the year 2024 was used, which includes imagery captured between 2021 and 2023 [1].

Each image includes RGB and NIR image bands and undergoes orthorectification using the Austrian Digital Elevation Model (DEM), precise ground control points, and GNSS-based acquisition point references. This results in high positional accuracy, ranging from 0.5 m to 1 m in flat and up to 5 m in steep mountainous terrain [108]. The spectral composition of image bands is shown in Table 3.2. Specific bounds may vary slightly between tiles, as different cameras (although of the same type) are used for individual imaging campaigns and BEV's processing algorithm can introduce minimal spectral shifts. As the introduced error is expected to be minimal, and the spectral mismatches between Sentinel-2 and orthophoto bands are much larger, this possible error source is disregarded during subsequent processing steps.

Table 3.2: Orthophoto spectral bands and their approximate wavelength ranges.

Band	**Wavelength Range** [nm]
Blue	420-500
Green	490-580
Red	580-690
NIR	690-880

3.1.3 Austrian Cadastral Data

Regardless of the image source, supervised DL applications require high-quality GT data that aligns well with its respective training images. Acquiring such datasets is a significant challenge and often a bottleneck in DL training. Governmental cadastral data, if available, can serve as a high-quality GT dataset for numerous applications. Cadastral agencies monitor land use and property boundaries, but the spatial accuracy and completeness of these datasets can vary based on legal definitions and governmental requirements [110]. In Austria, the responsibility for managing and processing cadastral data collected by authorised surveyors lies with BEV [111] and the data is stored in the publicly accessible two-dimensional cadaster system Digital Cadastral Map (German: Digitale Katastralmappe) (DKM).

Although Austrian cadastral data accuracy depends on the legally defined requirements of the collection period, and thus can vary between the cm and cm-to-m range, the DKM can be considered a complete and reliable dataset. It features a wide array of cadastral-related attribute classes which are commonly present in Austria (see Table 3.3). These classes encompass different categories, such as buildings, roads, water surfaces, and agricultural and forested areas. The DKM is updated daily and published via BEV's download centre semi-annually (once in April and once in October) [112].

The DKM is published in various vector formats, such as ESRI Shapefile, AutoCAD DXF, and GeoPackage [113]. DXF and Shapefiles are provided individually for each state in its corresponding MGI/Austria Gauss-Krüger zone, but are misaligned with aerial image footprints and not range-requestable, leading to potential processing problems. The GeoPackage version covers all of Austria and is range-requestable [114] but is published in MGI/Austria Lambert CRS (EPSG:31287), causing mismatches between the image and the cadastral data due to different CRS. As the benefits of the cloud-optimized GeoPackage format outweighs possible issues with CRS transformations, which can be addressed during processing, the GeoPackage datasets from 2021 to 2023 were used during this thesis [115, 116, 117].

Table 3.3: English translation of categorisation of land use codes in the Austrian cadaster.

Category	Code	Subcategory
Building areas	41	Buildings
	83	Adjacent building areas
Water body	59	Flowing water
	60	Standing water
	61	Wetlands
	64	Waterside areas
Agricultural	40	Permanent crops or gardens
	48	Fields, meadows or pastures
	53	Vineyards
	57	Overgrown areas
Forest	55	Krummholz
	56	Forests
	58	Forest roads
Other	42	Car parks
	62	Low vegetation areas
	63	Operating area
	65	Roadside areas
	72	Cemetery
	84	Mining areas, dumps and landfills
	87	Rock and scree surfaces
	88	Glaciers
	92	Rail transport areas
	95	Road traffic areas
	96	Recreational area
Gardens	52	Gardens
Alpine pasture	54	Alpine pasture

3.1.4 Used Software

Several different pieces of software were used for this thesis. The experiments were conducted using Python, with different packages for various processing steps. Data management was achieved with QGIS and geospatial Python extensions such as *GDAL*, *rasterio*, *fiona*, *geopandas*, and *cubexpress*. DL workflows were implemented with *PyTorch*, *segmentation-models-pytorch (smp)*, *Pytorch Image Models (timm)*, and *Weights & Biases (wandb)*. Furthermore, the Large Language Model (LLM) *ChatGPT* was used for code debugging and suggesting writing improvements and the proofreading software *grammarly* was relied on for correcting spelling and punctuation.

3.2 Data Download and Processing

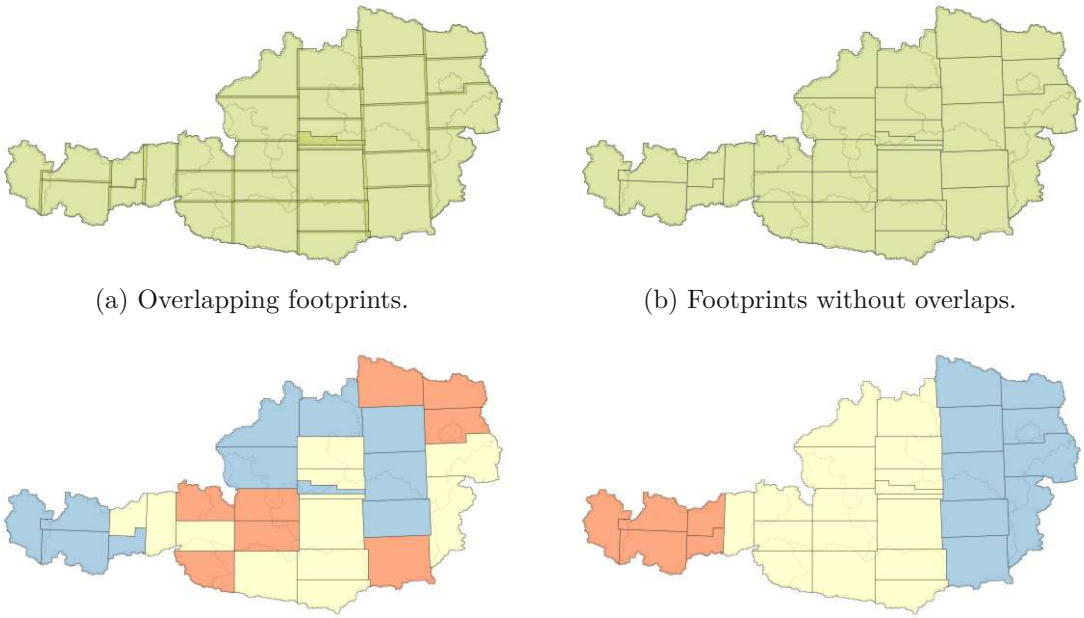
This section details the creation of the dataset used to train building delineation models and evaluate SR performance. Existing benchmark datasets for SR (see Section 2.2.3) are often tailored to specific models or tasks, limiting their suitability for an independent comparison. To ensure a fair evaluation, a new dataset was assembled using temporally and spatially aligned data from three sources: orthophotos as the HR baseline, cadastral data as the GT reference, and Sentinel-2 imagery as the LR input for SR or interpolation. While the dataset focuses on Austria, the workflow can be adapted to other regions and data sources for future downstream SR evaluations.

3.2.1 Matching of Orthophoto and Cadastral Data

By combining HR aerial imagery and cadastral data, it is possible to create reliable and accurate DL datasets that overcome many of the limitations associated with satellite-based image and GT data sources. As part of this thesis, a processing and downloading pipeline was developed and published as a Python package: *austriadownloader*, accessible in the PyPi package manager `https://pypi.org/project/austriadownloader/`. A detailed description of the workflow of this package was published in the proceedings of the AGIT 2025 conference [99]. The *austriadownloader* package will be used to download and process the orthophotos and cadastral data used in this thesis.

3.2.1.1 Pre-Processing: Spatial and Temporal Alignment

To ensure a seamless and unambiguous dataset, where each location has only one corresponding orthophoto and cadastral reference, a spatial Lookup Table (LUT) was developed to eliminate overlapping image footprints, temporally align the data sources, and handle CRS transformations.



(a) Overlapping footprints.

(b) Footprints without overlaps.

(c) Footprints with corresponding acquisition year; Red: 2021, Beige: 2022, Blue: 2023.

(d) Footprints with corresponding CRS EPSG; Red: 31254, Beige: 31255, Blue: 31256.

Figure 3.1: Orthophoto footprints for Austria with state boundaries.

Overlapping areas were removed from aerial imagery footprints (see Figure 3.1b) and the unique case of the aerial image of Windischgarten from 2023, which was partially redone and fully enclosed within its predecessor from 2022, was included as well. This allows the package to easily access the correct aerial image data sources for each location without any ambiguities and uncertainties. If an area near a dissolved overlap border is sampled, image data can contain `NoData` values, which can either be removed or flagged during dataset creation.

To temporally align the data sources (see Figure 3.1c for acquisition dates per year for aerial image tiles), each image was assigned to the closest preceding cadastral reference date, guaranteeing that all orthophotos have a corresponding temporal cadastral representation. As the image data is available in three different local CRS (see Figure 3.1d) and the cadastral data is in a single Austria-wide CRS, a transformation from the cadastral's CRS to the orthophoto's CRS-zone was implemented during the download process, guaranteeing a high degree of spatial accuracy in the combined dataset.

3.2.1.2 Download and Processing

To download aerial image and cadastral data, a sampling grid was created, indicating at which spatial locations image tiles and cadastral data should be acquired. This sampling grid is generated by intersecting a regular uniform grid with Austria's border polygon. The grid size is determined by the required output image height and width, both of which

are 512 pixels each. These prerequisites resulted in a sampling grid with 51,186 entries, which cover all of Austria and provide the spatial reference for downloading image data. To facilitate easier transformations between CRS and allow the querying of the created LUT, the centroid of each grid cell is saved together with a unique tile identifier.

The data processing pipeline comprises three main steps: first, window creation; second, raster download and processing; and third, vector download and rasterisation. Figure 3.2 provides an overview of the download process for a single pair of orthophoto image and cadastral mask. Output files are georeferenced and can be used directly in DL or other geospatial workflows. The *austriadownloader* pipeline was configured with the following options:

- Sample point positions: The centroids of each image patch from the created sampling grid are provided here.
- Cadastral code: All cadastral codes from Table 3.3 are provided, which will result in the download of a multi-class mask containing every cadastral class. This selection was made to validate the results of the downloading process and aid the identification of possible `NoData` areas.
- Pixel size in m: The GSD of both training and GT image is defined at 2.5 m to match the common spatial resolution of SR models.
- Image size in pixels: Output image width and height at 512 pixels.
- Image bands: Both RGB and NIR are selected to capture the full range of spectral bands available.

First, for each sampled centroid, the pipeline determines a bounding box using the provided parameters. The internal LUT identifies which MGI/Austria Gauss-Krüger zone the centroid occupies to ensure the correct orthophoto is retrieved. Additionally, the corresponding temporally matched cadastral reference is selected. Next, only the necessary raster subset is requested from the relevant COG, leveraging its internal overviews to match the user's chosen resolution. In parallel, cadastral GeoPackage data is downloaded with the *fiona* Python package, which allows streamable access and limits the required data volume to the corresponding orthophoto's spatial extent instead of the whole cadastral file. Next, all the downloaded vector data is filtered by the selected cadastral codes, merged, and transformed to the orthophoto's CRS, and finally rasterised into a segmentation mask with the assigned pixel value corresponding to the cadastral code, and the same resolution and extent as the orthophoto patch. The selection of building-labelled pixels to create binary masks for the building delineation is achieved in the DL model's dataloader itself. Cadastral masks and orthophoto subsets are saved as GeoTIFFs, ensuring pixel-perfect alignment.

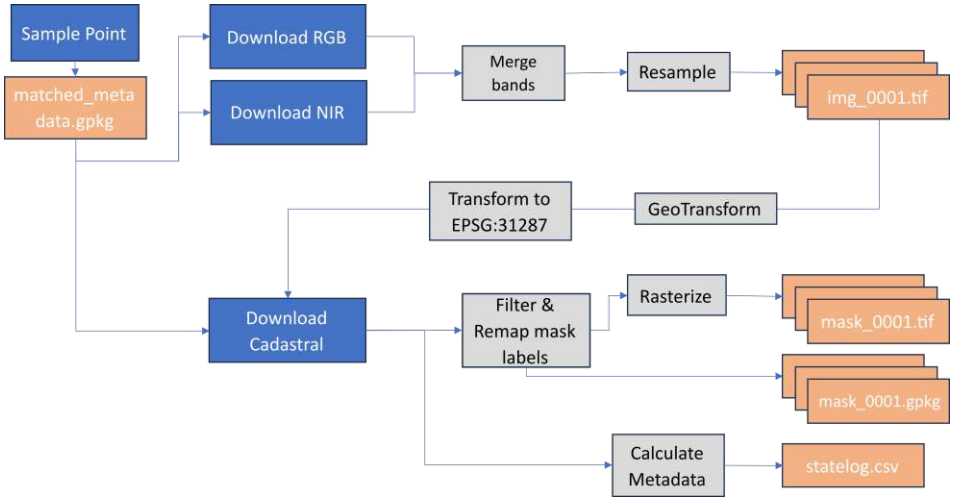


Figure 3.2: Flowchart for an iteration of the data processing applied with the *austriad-ownlaoder* package.

3.2.2 Matching and Download of Sentinel-2 Images

So far, the created dataset contains HR aerial images and the corresponding GT cadastral masks. To further complement this dataset the matching LR Sentinel-2 images need to be added. The download of Sentinel-2 images includes two distinct steps: gathering metadata for all viable images and then downloading the most suitable one. This was achieved by using the *cubexpress* package to access images from the data catalogue of harmonised Sentinel-2 L2A images hosted by GEE [105]. As there is a large number of Sentinel-2 tiles covering Austria, specific requirements and specifications were defined to select the most temporally and spectrally aligned tiles possible. Thus, for the initial selection of Sentinel-2 images, they should adhere to the following conditions. Images should:

- contain few or no clouds and cloud-shadows per image (expressed as a high Cloud Cover Score (CSS)).
- have few pixels with `NoData` values.
- be temporally aligned with the other data sources (orthophoto and cadastral).
- have a high degree of harmonised band-wise correlation to corresponding orthophotos as an additional guarantee of securing temporal and spectral alignment.

Step 1: Gathering metadata

The selection of Sentinel-2 tiles was achieved mainly by referring to meta- and image-data from the previously selected orthophotos. In this step, no images are downloaded, but metadata is gathered to allow the download of selected Sentinel-2 images in the next step. To access Sentinel-2 image metadata, a temporal filter is applied to match the acquisition period of the orthophotos. Since the orthophoto's acquisition time varies by tile, ranging from a few days up to several months, a minimum window of 100 days is defined, spanning 50 days before and 50 days after the orthophoto's mean acquisition date. If the orthophoto acquisition period exceeds this window, it is used instead of the 100-day period. This extended time frame ensures a sufficiently large pre-selection of Sentinel-2 images, which is later refined to a smaller set of well-matched ones.

Next, the time period of each sample is used to query all Sentinel-2 tiles in the given spatial location, which are sorted first by CSS and then by absolute time difference to the mean capture day of the orthophoto. This prioritises less cloudy images over recent ones, as cloud cover would prohibit its use for building delineation, while temporal offsets do not. Finally, the metadata (including download links) for the eight most suitable Sentinel-2 images was saved into a dataframe. No hard constraints were included in this process, which guaranteed at least one image per sample location, even if it contained clouds and was temporally misaligned. Such non-fitting samples can be filtered out in later processing steps.

Step 2: Selecting and downloading images

The download process is divided into sub-steps and parallelised by grouping Sentinel-2 images from the same spatial location into batches of up to eight images. The following describes the procedure for a single batch.

First, the orthophoto and cadastral mask for the target location are loaded into memory. Both are reprojected to the Sentinel-2 CRS, and spatial properties such as extent, width, and height are extracted. These parameters are used to construct spatial queries that retrieve the corresponding Sentinel-2 imagery from the GEE catalogue, enabled by its adherence to the COG standard. This ensures pixel-level alignment between Sentinel-2 and orthophoto images despite differences in resolution and CRS.

The queried Sentinel-2 images are then downloaded to a temporary directory. For each Sentinel-2 image, a binary `NoData` mask is generated. Additional `NoData` masks are created for the orthophoto and cadastral mask. These masks are resampled with NN interpolation to a common extent: Sentinel-2 from 10 m to 2.5 m, and orthophoto and cadastral masks from 2.5 m to 10 m. `NoData` values may occur for different reasons depending on the dataset. For orthophotos, they can result from sensor or processing failures, or when the query extent lies outside Austrian territory or beyond the bounds of an individual orthophoto tile. For cadastral masks, `NoData` values appear only when the query extent extends beyond Austrian territory. In Sentinel-2 imagery, they may occur due to sensor or processing failures, as well as from the application of cloud masking.

Next, each Sentinel-2 image is evaluated against the orthophoto to select the best match. First, the resampled masks (Sentinel-2, orthophoto, and cadastral) are merged to include all present `NoData` regions. The combined mask is applied during Cumulative Probability Distribution Function (CDF)-based histogram matching to exclude any pixel which contains `NoData` in any of the three image sources. This ensures that `NoData` pixels do not affect the harmonisation process. Correlation is then computed blockwise (16×16 pixels) rather than globally, yielding localised correlation values. The lowest 10th percentile of blockwise correlations, together with the overall proportion of `NoData` pixels, is used to rank the Sentinel-2 images for suitability. The image with the lowest `NoData` and highest correlation is retained and stored with its metadata, while the remaining batched Sentinel-2 images are discarded. This approach enables image selection based on spectral similarity, rather than temporal proximity alone.

During later dataset stratification, all tiles containing `NoData` values were excluded to avoid potential errors in SR models. As this decision was made after developing the downloading algorithm, and since the handling of `NoData` masks does not interfere with selection, the algorithm itself was left unchanged. This also preserves its applicability for datasets where strict `NoData` filtering is not required.

3.2.3 Stratification

Dataset stratification is an essential pre-processing step to adapt the data to user requirements and ensure compatibility with later DL algorithms. Stratification is performed using metadata generated during dataset creation with the *austriadownloader* package, which produced 51,186 tiles of matched orthophoto, and cadastral masks, and the subsequently downloaded Sentinel-2 images (see Figure 3.3a for the spatial distribution of all image tiles).

Based on these images, two separate datasets were created. The first, **Dataset Full**, was filtered only on `NoData` values and the presence of buildings in the cadastral mask. The second, **Dataset Filtered**, applied the same criteria with an additional filter, removing images with a blockwise correlation below 0.5. This choice was based on experiments indicating that **Dataset Full** contained more data than necessary, leading to longer training times with no improvement in performance. Instead of randomly discarding tiles, the correlation filter was applied to retain a sufficiently large dataset while excluding images likely affected by temporal and spectral offsets or other quality issues. Since both datasets follow nearly identical processing pipelines, they are described together, with the additional filtering of **Dataset Filtered** noted separately. Figure 3.3b shows the spatial distribution of **Dataset Full**, and Figure 3.3c shows the spatial distribution of **Dataset Filtered**. A summary of the number of remaining tiles after each processing step is given in Table 3.4.

First, all tiles containing `NoData` values in any image source (orthophoto, cadastral mask, or Sentinel-2) are removed. Such tiles can impact DL performance, as different SR model architectures handle `NoData` values inconsistently. For **Dataset Filtered**,

the correlation filter is then applied. Next, images without buildings are removed from both datasets to avoid over-representation of non-building areas (mainly forested or mountainous regions), which would increase data size and processing time while possibly lowering model performance. To maintain an even class balance and allow models to learn from non-building areas, a small subset of the removed tiles is reintroduced (as many as are required to compose 5% of images in each dataset). The reintroduction is based on random sampling, and since each dataset uses a unique random seed, the returned tiles differ between **Dataset Full** and **Dataset Filtered**.

Table 3.4: Processing steps and number of image tiles included in each step for both datasets.

Processing Step	Dataset Full	Dataset Filtered
Original	51,186	51,186
`NoData` Filter	46,189	46,189
Correlation threshold		24,943
Removal of non-building images	37,512	21,532
Re-adding of non-building images	39,486	22,665

(a) Unprocessed dataset. (b) **Dataset Full** (c) **Dataset Filtered**

Figure 3.3: Spatial representation of images remaining after dataset filtering.

The training of building delineation models during the conducted experiments requires the datasets to be split in training, validation, and testing datasets. Because the distribution of building pixels between image tiles is highly skewed, an additional stratification is performed to ensure a balanced representation of uninhabited, rural, and urban areas across these subsets. Without this step, delineation models might fail to generalise, as urban areas are strongly under-represented and need to be evenly distributed across subsets to mitigate this influence. The additional stratification is achieved by classifying the images based on the number of building pixels present, an attribute available in the pre-computed metadata of *austriadownloader*, allowing fast processing without re-loading each tile. An initial manual definition of class boundaries was attempted, but since they were influenced by subjective classification, an algorithmic method was adopted.

The Jenks Natural Breaks algorithm [118] was chosen to determine class boundaries because of its ability to identify natural groupings within the data. After experimenting with different class counts, five classes were selected as the optimal input for the Jenks algorithm. The resulting classification, which grouped images according to their percentage of building pixels, corresponded well with observed settlement patterns. Class boundaries for **Dataset Full** are listed in Table 3.5, and for **Dataset Filtered** in Table 3.6. The spatial distribution of classified datasets across Austria is illustrated in Figures 3.4a and 3.4b. Six total classes were defined:

- One manually defined class for images with exactly zero building pixels (the reintroduced non-building images), ensuring they do not distort the Jenks classification.
- Five classes derived from the Jenks algorithm, capturing settlement patterns across Austria.

(a) Jenks classes for **Dataset Full** (b) Jenks classes for **Dataset Filtered**

Figure 3.4: Classification of images by the Jenks Natural Breaks algorithm. The shade of blue describes to which class each image belongs, with class breaks presented in Table 3.5 and Table 3.6.

Table 3.5: **Dataset Full** class breaks based on building pixel percentages.

Class	Building Pixel Range (%)	Tile Count	Dataset Distribution (%)
0	0.00	1,974	5.00
1	0.00 – 1.33	29,509	74.73
2	1.33 – 4.52	6,051	15.32
3	4.52 – 10.84	1,445	3.66
4	10.84 – 23.50	449	1.14
5	23.50 – 52.90	58	0.15

Table 3.6: **Dataset Filtered** class breaks based on building pixel percentages.

Class	Building Pixel Range (%)	Tile Count	Dataset Distribution (%)
0	0.00	1,133	5.00
1	0.00 – 1.62	16,606	73.27
2	1.62 – 5.24	3,568	15.74
3	5.24 – 11.83	964	4.25
4	11.83 – 24.65	348	1.54
5	24.65 – 52.90	46	0.20

After the classification is achieved, both datasets are split into training (70%), validation (15%), and testing (15%) subsets in preparation for the subsequently performed building delineation. The class distribution defined by the Jenks classifications are preserved throughout this split, with the final distributions for **Dataset Full** shown in Table 3.7 and for **Dataset Filtered** in Table 3.8.

Table 3.7: **Dataset Full** class distribution across dataset splits.

Class	Whole Dataset (39,486)	Train (27,640)	Test (5,923)	Validation (5,923)
0	1,974 (5.00%)	1,382 (5.00%)	296 (5.00%)	296 (5.00%)
1	29,509 (74.73%)	20,656 (74.73%)	4,426 (74.73%)	4,427 (74.74%)
2	6,051 (15.32%)	4,236 (15.33%)	908 (15.33%)	907 (15.31%)
3	1,445 (3.66%)	1,011 (3.66%)	217 (3.66%)	217 (3.66%)
4	449 (1.14%)	314 (1.14%)	67 (1.13%)	68 (1.15%)
5	58 (0.15%)	41 (0.15%)	9 (0.15%)	8 (0.14%)

Table 3.8: **Dataset Filtered** class distribution across dataset splits.

Class	Whole Dataset (22,665)	Train (15,865)	Test (3,400)	Validation (3,400)
0	1,133 (5.00%)	793 (5.00%)	170 (5.00%)	170 (5.00%)
1	16,606 (73.27%)	11,624 (73.27%)	2,491 (73.26%)	2,491 (73.26%)
2	3,568 (15.74%)	2,497 (15.74%)	535 (15.74%)	536 (15.76%)
3	964 (4.25%)	675 (4.25%)	145 (4.26%)	144 (4.24%)
4	348 (1.54%)	244 (1.54%)	52 (1.53%)	52 (1.53%)
5	46 (0.20%)	32 (0.20%)	7 (0.21%)	7 (0.21%)

3.2.4 Dataset Quality Description

Since the created dataset is the foundation for evaluating SR models, describing its quality is essential. The primary concern is the spatial and pixel-wise alignment of the different data sources. For the orthophoto and cadastral GT data, this alignment is guaranteed: both originate from the same data provider BEV, where pixel-level matching is achieved through harmonisation procedures. Spatial accuracy is provided also by BEV, with the orthophoto achieving positional accuracies of 0.5 m to 1 m in flat terrain and up to 5 m in mountainous regions, while cadastral data is surveyed and typically ranges from cm to m-level precision.

In contrast, verifying alignment between Sentinel-2 imagery and the GT dataset is far more challenging. The native 10 m resolution of Sentinel-2 prevents a reliable pixel-wise comparison, and although L2A products (since March 2021) achieve absolute geolocation accuracies below 6 m, this still introduces uncertainty [119]. Consequently, Sentinel-2 pixels may be shifted by several meters relative to the GT data. This misalignment cannot be corrected and therefore represents an inherent error source that propagates to any super-resolved images derived from Sentinel-2. This limitation must be acknowledged when interpreting building delineation performance on super-resolved Sentinel-2 data. Visual inspection of the created dataset revealed no systematic misalignments, suggesting that the proposed processing pipeline provided sufficiently accurate data for evaluation.

3.3 Super-Resolution Inference

To compare SR algorithms against each other, each model must be applied to the downloaded Sentinel-2 images to generate HR outputs. While this process may appear straightforward, considerable effort was required to select, access, and implement the available SR models, further complicated by the proprietary nature of SR models developed by private companies.

3.3.1 Selection of Viable Super-Resolution Models

Many SR implementations exist, differing in the availability of pre-trained weights, achievable target resolution, applicability to Sentinel-2 imagery, the ability to super-resolve all 10 m spectral bands, and support for single-image inference. To maintain a fair comparison of SR models in this thesis, selection criteria were defined. Although this limited the number of eligible models, it ensured that the resulting comparisons were meaningful and methodologically consistent. These criteria which have to be met are listed here:

- Use of Sentinel-2 imagery, real or synthetic, for training the model. This ensures the applicability of the model to the generated evaluation dataset presented in Section 3.2.

- Partial public availability of model code and weights. Training SR models from scratch if these constraints are not met is not viable, as the large amounts of required computational resources would have exceeded the scope of this thesis. Published models need to be accessible for applying inference in one of these ways:

  1. Code and weights are publicly available, and inference with the provided materials and scripts is possible.
  2. Code and weights are available upon request to the authors, and inference is possible.
  3. The authors do not provide code or model weights, but agree to conduct the model inference on a provided dataset themselves and share the results.

- SR output resolution in the range of 2.5 m to 5 m. As a substantial proportion of available models super-resolves imagery to the output resolution of 2.5 m, any model output with a different resolution will be bicubically resampled to 2.5 m. Although this method might change each model's performance, it is a necessary step to ensure their comparability, and is also implemented similarly in other studies [4].

- Models need to super-resolve at least the Sentinel-2 image bands B02, B03, B04, and B08, covering the RGB and NIR spectrum. Feature extraction tasks focused on human-made structures can benefit from the use of spectral bands such as NIR [120], which captures vegetation and urban areas. The RGB and NIR image bands, with a native resolution of 10 m, will be used during the later building delineation.

Thus, a large number of SR models are excluded from the evaluation, as they do not meet the defined criteria. Models were excluded because not all components, model architecture, and weights are publicly available, prohibiting their application. This constraint excluded the models HighResNet [37], RResNet [16], SEN4x [12], Systema [63], EO-Research [17], and S2DR3 [121]. Additionally, DSEN2 [6] was not used as it only super-resolves its 20 m and 60 m bands to 10 m. Satlas [2] and SuperImage [122] could not be included as they super-resolve only RGB bands, and do not include NIR at all. Coding issues were encountered while implementing DiffFuSR [5], and L1BSR [9] could not be implemented because it relies on Level-1B (L1B) and L1C processed Sentinel-2 images, which were not included in the dataset used.

On the other hand, the models presented in Section 2.2.2 meet the defined criteria, which allows their application for this thesis. These include the models SR4RS [13], Evoland [8], DeepSent [40], Swin2Mose [14], LDSR-S2 [10], and two of the SEN2SR model variants: SEN2SR-Lite with the CNN backbone and SEN2SR-RGBN based on Mamba [11].

3.3.2 Applying Inference

A primary resource for running SR inference is the *superIX* framework [123]. It provides streamlined inference of multiple models by offering pre-trained and compiled implementations, thereby facilitating direct comparison of different SR architectures. As not all models super-resolve to the same resolution, the respective results were bilinearly resampled from the output resolution to 2.5 m (as seen in Figure 3.5).

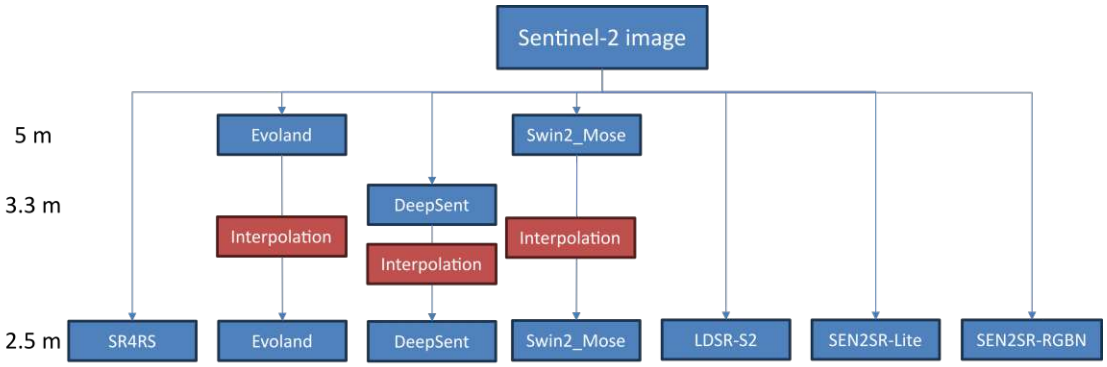


Figure 3.5: SR inference of SR models with target output resolutions and applied interpolations.

Super-resolving images with the **SR4RS** model was achieved with the *superIX* framework. Due to the nature of the model architecture, inference output was not provided in the native input shape of $4 \times 128 \times 128$ but as a cropped image of shape $4 \times 96 \times 96$. As these differing image dimensions would have rendered a comparison with other SR models impossible, a buffering of the input image was implemented. This increased the input dimension to $4 \times 144 \times 144$, which forced the model to provide outputs in the required super-resolved dimension of $4 \times 512 \times 512$. As the model does not expect a buffered border, artefacts are produced at the edge of the super-resolved image (see Figure 3.6), which may limit its applicability in downstream tasks, but is necessary to allow comparability.

Applying the **Evoland** and **Swin2Mose** model was achieved with the *superIX* framework, followed by a bilinear interpolation with the factor of 2 to retrieve results at the resolution of 2.5 m.

The **DeepSent** architecture required additional pre-processing, as it introduces a unique inference protocol. Instead of processing a single multi-band image, the input data must be organised in a specific directory structure (as seen in Listing 3.1). This approach greatly increases storage demands, as each image has to undergo sub-pixel shifts (to simulate multi-image SR) and is further split by each spectral band. Consequently, a single image is expanded into 108 separate files. In addition, the dataloader provided in the author's repository only accepts *png* input, which discards all georeferencing information. After the inference was conducted, the results were bilinearly interpolated to the resolution of 2.5 m.

Listing 3.1: DeepSent structure

```
dataset/
|-- lr_image_00001/
|   |-- b1/
|   |   |-- lr/
|   |   |   |-- lr_00.png
|   |   |   |-- lr_01.png
|   |   |   |-- lr_02.png
|   |   |   |-- lr_03.png
|   |   |   |-- lr_04.png
|   |   |   |-- lr_05.png
|   |   |   |-- lr_06.png
|   |   |   |-- lr_07.png
|   |   |   `-- lr_08.png
|   |-- b2/
|   |   `-- ...
|   `-- .../
|       `-- ...
|-- lr_image_00002/
|   `-- ...
`-- ...
```

The inference of the **LDSR-S2** model was initially performed using the *superIX* framework. As the authors further developed and fine-tuned their model without publishing updated weights on the *superIX* platform, I contacted them and they agreed to prepare

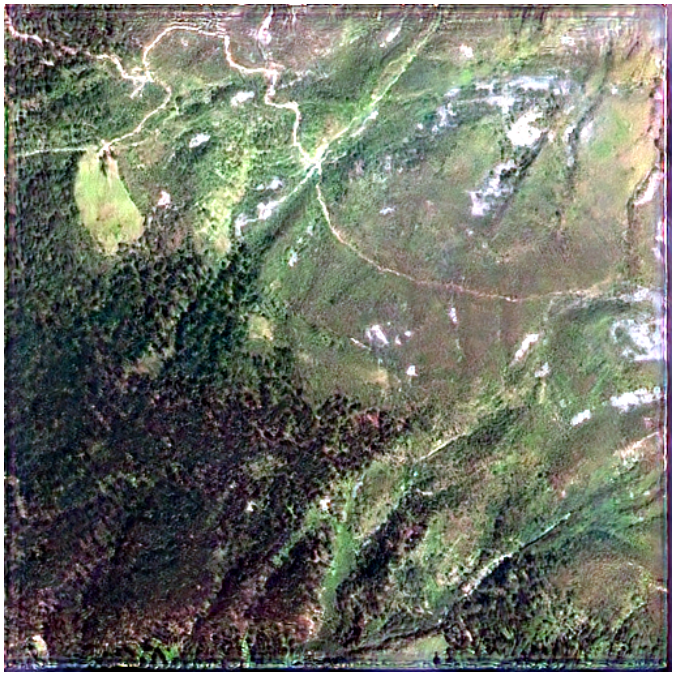


Figure 3.6: Example of artefacts at the border of an inferred image by SR4RS.

the inferred dataset. For this purpose, the created datasets were transferred to their server infrastructure, where the Graphics Processing Unit (GPU)-intensive inference was conducted. Owing to the iterative nature of the diffusion-based LDSR-S2 architecture, inference required considerably more time than traditional approaches. After completion, the super-resolved images were transferred back to local storage.

Both SEN2SR models, **SEN2SR-Lite** and **SEN2SR-RGBN**, were inferred with provided scripts. **SEN2SR-Lite** required no specific setup, and inference was conducted as per the instructions of the authors. As **SEN2SR-RGBN** is built with Mamba and depends on GPUs with CUDA versions above 12, a special setup had to be arranged. The authors arranged for an *ssh* connection to GPUs equipped with the necessary hardware and thus enabled the inference with this model.

3.3.3 Super-Resolution of Proprietary Models

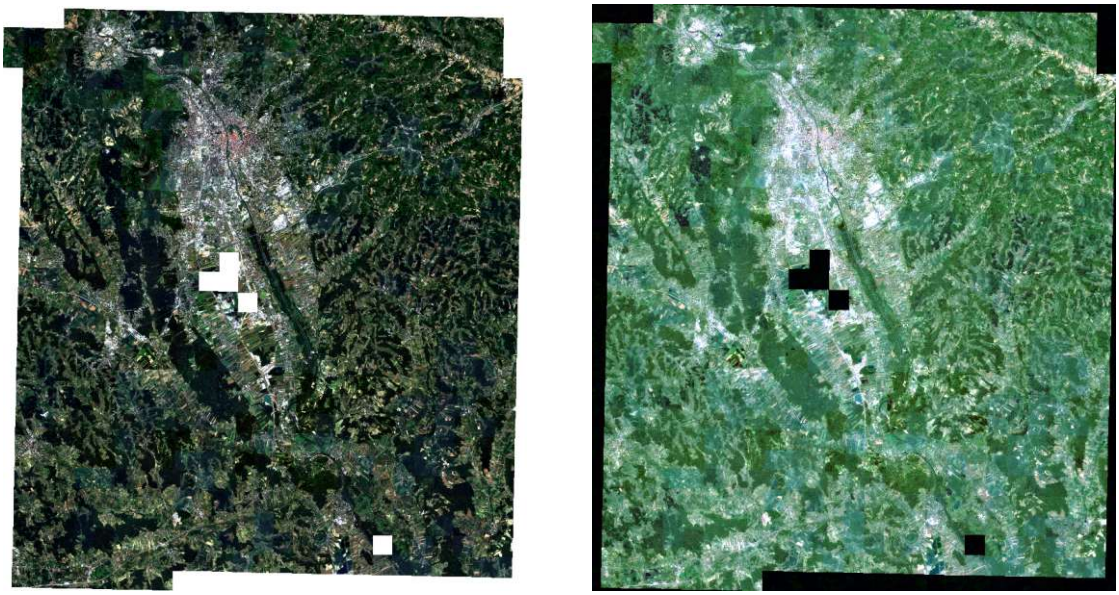
Not all SR models are developed by public research institutes; some originate from commercial companies. While such providers may supply sample scripts for limited inference, the underlying code and model weights are usually not publicly available. Nevertheless, evaluating proprietary models remains relevant, as they may deliver reliable results at lower cost and higher temporal resolution compared to HR sensors. To access such models, I contacted the authors of Tracasa [15] and S2DR3 [121] directly to obtain API keys or to arrange inference on supplied imagery. Both indicated that high server costs restricted the amount of data they could process, and company policy prevented the sharing of model weights. Thus, I created a smaller evaluation dataset suitable for inference with their services.

This subset was designed to capture the diversity of Austrian landscapes, including non-populated, rural, and urban areas. To ensure variety while reducing the impact of `NoData` values and to meet the inference requirement of Ayala et al. [15] for merged input patches, the city of Graz, Steiermark, and its surrounding suburbs were selected as the evaluation region. From **Dataset Full**, a subset of 700 Sentinel-2 images was extracted (see Figure 3.7a), with the corresponding class distribution summarised in Table 3.9. Compared to **Dataset Full** (see Table 3.5), this subset over-represents urban areas, which is beneficial as they are the main area of interest for building delineation models and thus allow a more comprehensive evaluation. By adhering to the training, validation, and testing splits defined in Section 3.2.3 only 59 images were assigned to the test subset.

As training an independent DL building delineation model on this small dataset was infeasible, two alternative evaluation strategies were explored: first, a qualitative analysis of inferred images; and second, transfer learning-based building delineation. The former provides only visual insights and does not allow conclusions about downstream SR performance. The latter may allow a rough performance estimate by applying a building delineation model, originally trained on bilinearly interpolated Sentinel-2 images, to the super-resolved ones. This approach can introduce tremendous biases, as the pre-trained

Table 3.9: Graz evaluation dataset with the corresponding class distribution.

Class	Building Pixel Range (%)	Tile Count	Dataset Distribution (%)
0	0.00 – 0.00	2	0.29
1	0.00 – 1.33	362	51.71
2	1.33 – 4.52	236	33.71
3	4.52 – 10.84	60	8.57
4	10.84 – 23.50	36	5.14
5	23.50 – 52.90	4	0.57



(a) Sentinel-2 images of the evaluation subset covering Graz. Empty areas are present due to images removed with the `NoData` filter.

(b) Inferred image of the Graz evaluation dataset by the Tracasa model. The inferred image displays a spectral shift.

Figure 3.7: Evaluation subset of Graz and the corresponding inferred Tracasa image.

model is influenced by how well the SR method retains Sentinel-2 image characteristics such as spectral distribution and spatial position. If a SR model introduces changes from Sentinel-2 and its bilinear interpolated images, the building delineation model will not be able to cope with them, as the model's pre-trained layers expect Sentinel-2-like features. Despite this limitation, the proposed method was used as a practical first evaluation proxy in the absence of other quantitative methods. To test the general applicability of this strategy, the same transfer-learning approach was also applied to the publicly available SR models presented in Section 3.3.1.

3.3.3.1 Tracasa

The inference of the model published by Tracasa (see Section 2.2.2.7) was achieved by providing the whole evaluation sub-dataset to the authors, who merged the images, filled `NoData` areas and performed the SR. No information about the inference procedure can be provided as it was not shared. The whole inferred image is shown in Figure 3.7b and includes a visible spectral shift compared to the Sentinel-2 images.

3.3.3.2 S2DR3

After initial contact with the authors was established, and an API-key for user-defined inference was provided, processing the proposed extensive capacity was not allowed as it exceeded the allocated server resources. Using the S2SR3 model would have provided interesting insights, as it was the only model super-resolving its output to the spatial resolution of 1 m. This could have further validated if the approach of interpolating super-resolved images to the uniform resolution of 2.5 m for comparing SR models is suitable. The limited access to S2DR3 reduces the comparison of proprietary models to just the model from Tracasa.

3.4 Building Delineation

With the datasets generated in Section 3.2, building delineation networks were trained on the orthophoto and super-resolved Sentinel-2 images. This enables an assessment of the viability of SR for downstream tasks and highlights performance differences between the outputs of individual SR models, interpolation methods, and the orthophoto baseline.

3.4.1 Setup Parameters

Selecting appropriate parameters for each DL experiment is crucial, as they directly influence the training process and define achievable model performance. A complete specification includes the dataset, model architecture, loss function, and evaluation metrics, all of which are required to design a model capable of performing building delineation. Based on the literature presented in Section 2.3, this section outlines the setup parameters considered for this experiment.

3.4.1.1 Datasets

As discussed in Section 3.2.3, two datasets were created: **Dataset Full**, which contains all images without `NoData` values, and **Dataset Filtered**, which applies an additional correlation-based filter to remove orthophoto and Sentinel-2 pairs with low blockwise correlation values under 0.5. To evaluate whether building delineation models benefit from larger or better-filtered datasets, experiments are conducted using the same model architectures on both datasets. The exact setups and their results are presented in Section 3.4.5.1 and Table 3.11.

Although the respective test subsets of **Dataset Full** and **Dataset Filtered** differ in their specific images, they underwent the same stratification process and share a similar class distribution. This enables a meaningful comparison between them, to determine which dataset is better suited for evaluating SR models. The evaluation of the datasets for subsequent experiments is based on the performance in terms of the evaluation metrics IoU and F_1.

3.4.1.2 Model Architectures

The literature review in Section 2.3 showed that a couple of DL architectures have been applied to building delineation. Most approaches rely on CNN-based models, such as UNet variants or more advanced architectures like HRNet, SAM, or diffusion-based models. To keep the experiments manageable, this thesis focuses on two representative model types: a basic UNet with different backbones and HRNet with an OCR head. Both implementations are described below with emphasis on their suitability for building delineation on medium-resolution remote sensing images.

In this work, the **UNet** architecture was constructed using the *smp* library, which enables efficient customisation, and *timm*, which provides access to advanced backbones. These frameworks simplify implementation compared to building models from scratch, but allow less fine-grained control than the original codebases. Based on preliminary experiments and prior experience, multiple backbones available in *smp* were tested, including HRNet and variants of ResNeXt with SE modules. All UNet models were set up with the parameters presented in Listing 3.2, the backbone (`encoder_name`) was defined depending on the conducted experiment:

Listing 3.2: UNet model setup

```
import segmentation_models_pytorch as smp

model = smp.Unet(
    encoder_name=config.model.encoder,
    encoder_weights=None,
    encoder_depth=5,
    decoder_channels=[512, 256, 128, 64, 32],
    decoder_use_norm="batchnorm",
    decoder_attention_type="scse",
    decoder_interpolation="bilinear",
    in_channels=4,
    classes=1,
    activation=None
)
```

HRNet was implemented using the original semantic segmentation repository provided by the authors in [124]. Several modifications were made:

- An OCR head was integrated, following the authors' recommended configuration, and adapted for binary segmentation.
- The model output was adjusted to produce full-resolution prediction maps. As HRNet normally outputs at a reduced resolution to save memory, a bilinear upsampling by a factor of two was added to match the input resolution.
- Since HRNet outputs both a main prediction and an auxiliary prediction map from intermediate stages, the loss function was adapted to incorporate both. Two model variants were trained: one using only the main output, and one combining main and auxiliary outputs.

HRNets were created with code presented in Listing 3.3, with the configuration options available in Listing 7.1:

Listing 3.3: HRNet model setup

```
from model_files.HRNet.lib.models import hrnet, seg_hrnet, seg_hrnet_ocr

config = OmegaConf.load('./model_files/HRNet/configs/seg_hrnet_ocr.yaml')
model = seg_hrnet_ocr.get_seg_model(cfg=config)
```

3.4.1.3 Loss Functions

Two loss functions were employed: Focal Tversky Loss (FTL) [125] and Binary Cross-Entropy (BCE) [126]. FTL (see in Equation 3.1) extends the standard Focal Loss by using the weights α, β, and γ, which allows the fine-tuning of loss calculation on imbalanced datasets. The standard values for these parameters are $\alpha = 0.7$, $\beta = 0.3$ (with $\alpha + \beta = 1$), and $\gamma = 0.75$. As the presented task of building delineation profits from a larger γ value than the standard value of 0.75, it was initialised with $\gamma = 1.33$ instead. During fine-tuning, the γ value was adjusted to better represent the imbalanced dataset used for experiments. Adjusting α and β would change how strongly False Positive (FP) and False Negative (FN) are penalised, while γ controls the focus on hard-to-classify pixels. This makes FTL well-suited for under-represented classes, such as building footprints.

$$\mathrm{FTL} = \left(1 - \frac{TP}{TP + \alpha \cdot FP + \beta \cdot FN}\right)^{\gamma} \tag{3.1}$$

$$TP = \sum (y_{\mathrm{true}} \cdot y_{\mathrm{pred}})$$
$$FP = \sum [(1 - y_{\mathrm{true}}) \cdot y_{\mathrm{pred}}]$$
$$FN = \sum [y_{\mathrm{true}} \cdot (1 - y_{\mathrm{pred}})]$$

BCE (see Equation 3.2) is widely used for binary segmentation. It computes the pixel-wise cross-entropy between prediction $\hat{y}_i$ and GT y_i, penalising misclassifications. While effective, BCE may perform poorly in imbalanced datasets.

$$\text{BCE} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{3.2}$$

y_i is the true label (GT)
$\hat{y}_i$ is the predicted probability
N is the number of samples

To leverage the advantages of both losses, a weighted combination was defined (see Equation 3.3), with initial weights $\eta = 0.5$ and $\xi = 0.5$. This combines pixel-level precision from BCE and class imbalance robustness from FTL.

$$\text{FTL\_BCE} = \eta \cdot FTL + \xi \cdot BCE \text{ , with } \eta + \xi = 1 \tag{3.3}$$

η weight for FTL
ξ weight for BCE

3.4.2 Fine-Tuning strategy

While selecting a dataset, model, and loss function is straightforward, identifying the combination of these setup parameters that yields the best performance is a challenging task. To address this, a fine-tuning strategy must be defined to evaluate the different selections of setup parameters. Since the goal of this thesis is to compare different SR algorithms, all experiments on super-resolved datasets must follow a similar setup. This requires training all models under exactly similar conditions: using the same dataset, model, and loss function. While this approach may not produce the best-performing building delineation model for each SR algorithm, it ensures their comparability, which is more important for this study.

Ensuring fair comparison requires similar parameters across all building delineation models, while also accounting for the need to use a high-performing setup. To balance these needs, the entire building delineation pipeline is fine-tuned on the orthophoto dataset. This dataset contains no hallucinations and aligns perfectly with the GT cadastral masks. Fine-tuning on orthophotos introduces some bias, as SR frameworks that resemble the orthophoto output might benefit. However, fine-tuning each model separately for every super-resolved Sentinel-2 dataset would create two problems: first, it would require high

amounts of computational power; and second, it would break comparability, since models would be optimised differently. The main drawback of orthophoto-based fine-tuning is that the unique spatial and spectral properties of the orthophoto sensor could influence setup parameter selection. Resampling from 0.2 m to 2.5 m resolution produces different characteristics than resampling or super-resolving Sentinel-2 data from 10 m to 2.5 m, which might affect model performance.

An alternative would be to fine-tune the delineation networks on interpolated Sentinel-2 images. This would eliminate sensor differences, but it would also compromise the experimental design. The interpolated datasets serve as a baseline for comparing SR methods. If they were used for fine-tuning, models might become biased towards interpolated data, undermining the validity of the comparison. Additionally, any limitations introduced by orthophoto fine-tuning would still carry over to both interpolated and SR datasets in the same way, preserving comparability. For these reasons, fine-tuning and hyperparameter selection are conducted on the orthophoto dataset, and only if required, they are validated on the interpolated datasets. Once the optimal configuration is identified, models are trained on the super-resolved Sentinel-2 datasets and compared for effectiveness.

3.4.3 Training Setup

Defining a uniform training setup is especially important when comparing the results from different models. For the experiments, similar methods for reading and processing data, setting up schedulers to automatically influence the model training, and logging results into a uniform database were used. These, and several more setup parameters, will be described in this section.

All training was achieved on an NVIDIA L40 GPU with 46 GB of Random-Access Memory (RAM). This allowed me to test models with large backbones and to run several models at the same time, drastically increasing experiment speed. All models were trained from scratch, and no pre-trained weights were used. While *torchgeo* [127] provides pre-trained weights compatible with Sentinel-2, they are only available for specific model configurations, which were not used during the experiments. The Adam optimiser was used with an initial value of $0,0001$.

To load super-resolved Sentinel-2 images and the corresponding masks into the building delineation model, a dataloader class was developed to handle all relevant operations. First, the super-resolved image is loaded and transformed into the `float32` format, which is required by the used DL framework *PyTorch*. No normalisation is implemented, but the images are cast to the range of 0 to 1, either during the previously applied SR inference or in the dataloader. As some models were trained on interpolated Sentinel-2 images (bilinear, NN, and bicubic), a function was implemented to load and directly interpolate the LR Sentinel-2 image into the higher spatial resolution of 2.5 m. This mimics the process of super-resolving the images, and while applying this step during runtime increases the loading time by a small amount, it reduces the required memory on the disk space. Next, the multi-class cadastral mask is loaded and converted to a

binary mask containing only building footprints. The created dataset contains images of shape $4 \times 512 \times 512$, which introduces long training times due to their large size. As they contain a high amount of redundant information, many areas of the images contain no building footprints at all, a sampling process was developed to select the most relevant portion of each image. This sampling is achieved by splitting the input image into four regular subsamples of shape $4 \times 256 \times 256$, and selecting the one with the highest amount of building pixels present. If no building pixels were present in any subsample, a random one, with a preset seed to keep the selection uniform across epochs, was selected. This reduced the training time of each epoch by about three-quarters, while retaining most of the relevant information. Additionally, methods are implemented to retain spatial transformation parameters during the application of the building delineation model. While this requires slightly more complex scripts, it allows the generation of georeferenced masks, which are essential for any task revolving around geospatial data.

The training process is supervised by applying extensive logging of model parameters and performance. Training and validation losses, the validation metrics presented in Section 3.4.1.3, epoch count, learning rate, and a sample of segmented building masks from the validation set are logged regularly. This is achieved by connecting the training process to the *wandb* API, which provides a graphical user interface with overviews of the provided metrics. Based on these logged losses, several callback functions are implemented. First, a learning rate scheduler checks the validation loss after every epoch and reduces it by a factor of 0.5 after five epochs with no improvements. Similarly, an early stopping callback ends the training process after ten epochs without improvements in the validation loss. At the end of each epoch, model weights are saved into local storage, overwriting the set of weights from the previous epoch. Additionally, the weights from the currently best-performing epoch are saved separately and later used to calculate the evaluation metrics on the test dataset. During validation and testing, a threshold of 0.75 is applied to the model's sigmoid output (between 0 and 1) to classify it into background and building pixels. To determine the threshold value, several incrementally increased values were applied to the test dataset, with 0.75 yielding the most promising results.

3.4.4 Evaluation Metrics

Unlike loss functions, evaluation metrics are not used during training, but instead, they assess model quality during validation and testing. They are standardised and allow for fair comparison across models from different publications. Formulas and definitions of the presented metrics were accessed from Terven et al. [128]. Two types of evaluation metrics were used, image-based and object-based metrics.

3.4.4.1 Image-Based Metrics

Because different metrics may produce conflicting results, a predefined ranking was established, with emphasis placed on IoU and F_1, and less focus placed on Accuracy and Specificity given the dataset imbalance. The presented metrics are listed in ascending

order of importance for evaluation, and for all metrics, higher values correspond to better model performance.

Accuracy measures the proportion of correctly classified pixels. However, Accuracy can be misleading in imbalanced datasets (e.g., predicting only background yields high scores):

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.4}$$

Specificity captures correct background classification:

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{3.5}$$

Recall measures the detection rate of foreground pixels:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.6}$$

Precision quantifies the proportion of predicted positives that are correct:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.7}$$

IoU measures segmentation overlap:

$$IoU = \frac{TP}{TP + FP + FN} \tag{3.8}$$

Analysed together, Specificity, Recall, and Precision give a more complete picture than individually. F_1 is calculated as the harmonic mean of **Precision** and **Recall**. In binary segmentation, F_1 is equivalent to the Dice coefficient.

$$\text{F}_1 = 2 * \frac{Precision \times Recall}{Precision + Recall} \tag{3.9}$$

As the datasets employed in this thesis include images which contain no buildings, specific adaptations to the evaluation metrics were required. In cases where the GT consists solely of background pixels and the model correctly predicts only background, the standard metric formulations would yield values of zero or break due to division-by-zero errors, as no True Positive (TP) values are present to produce a non-zero IoU or F_1. To address this, an additional conditional check was introduced to enforce that the metrics are assigned values of IoU= 1 and $F_1 = 1$ when the model correctly predicts an entirely background image. This adjustment implicitly favours conservative predictions in regions

where no buildings are present. While such behaviour may be considered a drawback in certain contexts, it ensures a more robust evaluation of models and reflects the challenges of real-world applications, where a perfectly stratified dataset containing only images with buildings present cannot be assumed.

3.4.4.2 Custom Object-based Metrics

The presented metrics are pixel-based and do not explicitly evaluate object-level performance. For building extraction, however, object-level evaluation can be important as well. To address this, custom metrics from the ESAOpenSR project were adapted [129]. While the presented metrics are not peer-reviewed, and their relevance for evaluating segmentation maps is not proven, they might provide additional information regarding the performance of building segmentation models. These metrics include:

1. **Object Prediction Average (OPA)**: Mean overlap ratio of predicted and GT reference objects.

2. **Object Found Average (OFA)**: Fraction of GT buildings correctly detected by overlap.

The metrics are calculated as follows: objects (all continuously labelled buildings) are extracted from both predictions and GT; overlaps above a 50% threshold are labelled as found. Overlap ratios and counts are then aggregated into OPA and OFA. Detected objects are further grouped by size in pixels (0-9, 10-19, 20-34, 35-49, 50-74, larger than 75) to analyze performance across scales. Importantly, these metrics do not penalise FP, meaning models that over-predict are not penalised.

3.4.5 Experiments to Define Suitable Setup Parameters

In accordance with the fine-tuning strategy defined in Section 3.4.2, building delineation model parameters are selected based on their performance on the orthophoto dataset. A first overview of all models used in this experiment is provided in Table 3.10. Each model, with the relevant setup parameters and its achieved metric results, will be shown in subsequent tables to allow for their detailed comparison. Due to the imbalanced dataset, the metrics Accuracy and Specificity do not provide much insight, as they get heavily influenced by the correct identification of background pixels, a task which all models achieve well. Thus, they are not presented in the following results. Figure 3.8 displays a graphical demonstration of the different setup parameters available.

Table 3.10: All models with an overview of their setup parameters.

Name	Dataset	Model	Model Backbone	Loss	Loss Parameters (for FTL $\alpha = 0.7$, $\beta = 0.3$)	Sensor
01	Full	UNet	se_ResNext50	FTL	$\gamma = 1.3$	Orthophoto
02	Filtered	UNet	se_ResNext50	FTL	$\gamma = 1.3$	Orthophoto
03	Full	UNet	HRNet	FTL	$\gamma = 1.3$	Orthophoto
04	Filtered	UNet	HRNet	FTL	$\gamma = 1.3$	Orthophoto
04_1	Filtered	HRNet	OCR	FTL	main loss, $\gamma = 1.3$	Orthophoto
04_2	Filtered	HRNet	OCR	FTL	main & auxiliary loss, $\gamma = 1.3$	Orthophoto
05	Filtered	UNet	se_ResNext50	BCE		Orthophoto
06	Filtered	UNet	se_ResNext50	FTL + BCE	$\gamma = 1.3$	Orthophoto
07	Full	HRNet	OCR	FTL	main & auxiliary loss, $\gamma = 1.3$	Orthophoto
08	Filtered	UNet	se_ResNext101	FTL	$\gamma = 1.3$	Orthophoto
09	Filtered	UNet	se_ResNext50	FTL	$\gamma = 1.0$	Orthophoto
10	Filtered	UNet	se_ResNext50	FTL	$\gamma = 1.5$	Orthophoto
11	Filtered	UNet	se_ResNext50	FTL	$\gamma = 1.0$	Bilinear Sentinel-2
12	Filtered	UNet	se_ResNext50	FTL	$\gamma = 1.3$	Bilinear Sentinel-2
13	Filtered	UNet	se_ResNext50	FTL	$\gamma = 1.5$	Bilinear Sentinel-2

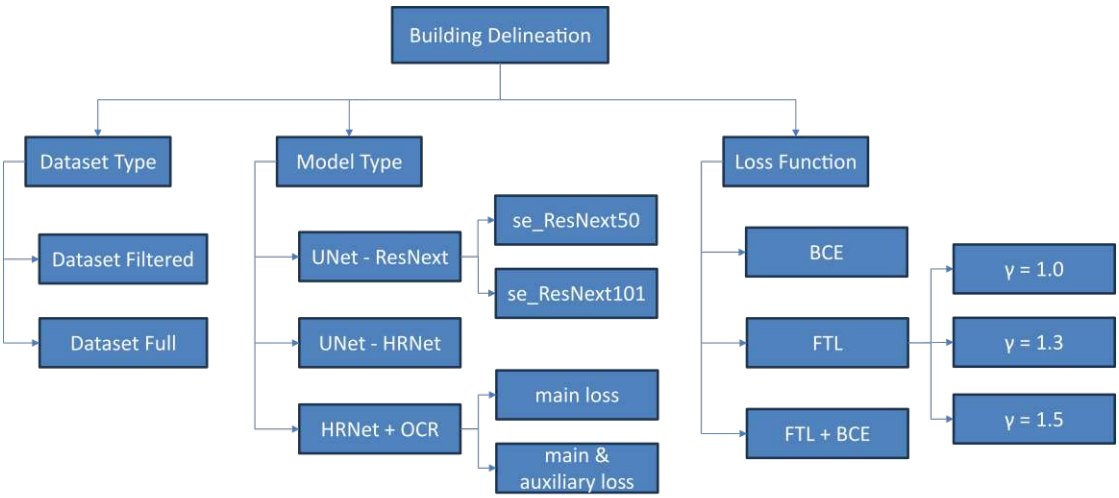


Figure 3.8: Overview of all available setup parameters.

3.4.5.1 Datasets

To determine whether **Dataset Full** or **Dataset Filtered** should be used for subsequent experiments, a comparative test was conducted. Both datasets were evaluated using different model architectures trained on orthophoto images. This comparison ensures that the choice of dataset does not systematically favour one model architecture over another. More details of the training setup are provided in Table 3.10.

The results, summarised in Table 3.11, show that models trained on **Dataset Filtered** outperform those trained on **Dataset Full** when evaluated on their respective test sets. The differences are small for the UNet with an HRNet backbone and the native HRNet, with 0.004 and 0.002 in IoU 0.007 and 0.005 in F_1, but an improvement is clearly evident in the UNet model with ResNext backbone. Its performance on IoU and F_1 amount to 0.014 and 0.015 respectively, which indicates a conclusive performance gain by switching to the smaller and better filtered dataset. Importantly, the reduction in training images does not degrade performance, since **Dataset Filtered** remains sufficiently large to enable models to learn generalizable patterns. Moreover, the smaller dataset reduces training time and resource consumption, which is a practical advantage.

Table 3.11: Models and their test results to determine which dataset, **Dataset Full** or **Dataset Filtered** will be used. The best results on the metrics IoU and F_1 are highlighted in bold.

Name	Dataset	Model	Precision	Recall	IoU	F_1
01	Full	UNet (ResNext)	0.766	0.611	0.538	0.666
02	Filtered	UNet (ResNext)	0.769	0.633	**0.552**	**0.681**
03	Full	UNet (HRNet)	0.756	0.610	0.535	0.661
04	Filtered	UNet (HRNet)	0.766	0.614	**0.539**	**0.668**
07	Full	HRNet	0.723	0.572	0.499	0.627
04_2	Filtered	HRNet	0.733	0.575	**0.501**	**0.632**

Although the removal of poorly correlated Sentinel-2 image pairs has no direct impact in this comparison, as the experiments here were conducted exclusively on orthophotos, it is reasonable to expect a positive effect when the same filtering strategy is applied to super-resolved Sentinel-2 datasets. By eliminating inconsistent training examples, **Dataset Filtered** likely provides images that will improve downstream model performance on super-resolved imagery.

3.4.5.2 Model Architectures

Next, to determine the most suitable model architecture for the task of building delineation, three sets of experiments were conducted with the selected model infrastructures. First, individual tests were carried out to establish the most performant setup for each model (see Table 3.12 for HRNet and Table 3.13 for UNet). Second, the best configurations of these models were compared against each other (see Table 3.14) to identify the most appropriate architecture for subsequent experiments. All models were trained on **Dataset Filtered** using orthophoto images.

To evaluate the optimal setup for HRNet, two model variants were tested: one trained

with only the main loss, and another trained with a combination of main loss (weight = 1) and auxiliary loss (weight = 0.4). Both used a OCR head and the FTL loss function, with additional parameters specified in Table 3.10. The results in Table 3.12 show almost identical performance across both setups, with differences between both models as small as 0.001 in both IoU and F_1. As the numerical results are inconclusive, I followed the recommendation of the public HRNet repository [124] and adopted the main & auxiliary loss approach for further experiments.

Table 3.12: HRNets and their results to determine which loss setup performs better. The best results on the metrics IoU and F_1 are highlighted in bold.

Name	Model	Loss Method	Precision	Recall	IoU	F_1
04_1	HRNet	main loss	0.723	0.582	**0.502**	**0.633**
04_2	HRNet	main & auxiliary loss	0.733	0.575	0.501	0.632

Next, an evaluation was conducted to determine which ResNeXt backbone is better suited for the UNet model. Larger backbones, such as se_ResNeXt101, contain more parameters, enabling stronger generalisation and robustness to noisy data. However, they do not necessarily yield better results for simple tasks such as binary segmentation, where smaller models may perform equally well or better while consuming fewer resources. To test this, two UNets were trained with se_ResNeXt50 and se_ResNeXt101 backbones. As shown in Table 3.13, the smaller se_ResNeXt50 achieves higher performance while requiring fewer computational resources, making it the more suitable choice for subsequent experiments.

Table 3.13: UNets and their results to determine which se_ResNext backbone performs better. The best results on the metrics IoU and F_1 are highlighted in bold.

Name	Model	Backbone	Precision	Recall	IoU	F_1
02	UNet	se_ResNext50	0.769	0.633	**0.552**	**0.681**
08	UNet	se_ResNext101	0.766	0.627	0.546	0.676

Finally, the best-performing configurations from the individual tests were compared directly. Additionally, a UNet with a pre-built HRNet backbone was included in the evaluation. The results in Table 3.14 indicate that both UNet variants outperform the original HRNet. Between the two UNets, the se_ResNeXt50 backbone performs better than the HRNet backbone, while also being more resource-efficient. Consequently, the UNet with se_ResNeXt50 is selected as the reference architecture for all further experiments.

Table 3.14: Different models and their results to determine which architecture performs best. The best results on the metrics IoU and F_1 are highlighted in bold.

Name	Model	Additional Information	Precision	Recall	IoU	F_1
02	UNet	se_ResNext50 backbone	0.769	0.633	**0.552**	**0.681**
04	UNet	HRNet backbone	0.766	0.614	0.539	0.668
04_2	HRNet	main & auxiliary loss	0.733	0.575	0.501	0.632

3.4.5.3 Loss Functions

The final experiment concerns the selection and fine-tuning of the loss function. As reviewed in Section 2.3, the literature reports the use of several different losses. To determine which is most effective on the presented dataset, I designed a series of experiments: first, comparing different loss functions; second, fine-tuning the most promising one; and finally, validating its performance on Sentinel-2 data to ensure its transferability from orthophotos. For all loss function-related experiments, a UNet with se_ResNext50 backbone is trained on **Dataset Filtered**.

First, three loss setups were tested: FTL (with a default parameter configuration), BCE, and a weighted sum of both (weighted at = 0.5 each). As shown in Table 3.15, FTL alone delivers the best results. The use of BCE decreases performance significantly, particularly when applied alone, though the negative impact is partially mitigated in the combined setup.

Table 3.15: Models trained with different loss functions and their results, to determine which combination performs best. The best results on the metrics IoU and F_1 are highlighted in bold.

Name	Loss Function	Loss configuration	Precision	Recall	IoU	F_1
02	FTL	$\alpha = 0.7$, $\beta = 0.3$, $\gamma = 1.3$	0.769	0.633	**0.552**	**0.681**
06	FTL & BCE	both weighted at 0.5; $\alpha = 0.7$, $\beta = 0.3$, $\gamma = 1.3$	0.783	0.597	0.531	0.664
05	BCE		0.818	0.497	0.463	0.601

To further refine the choice, I tested variations of the γ parameter in FTL, as it directly influences the weighting of hard-to-classify pixels, a critical aspect for imbalanced datasets with a particular focus on accurately reconstructing building footprints. Other parameters, α and β, were not tested further, as this would have dramatically increased the number of required experiments. Three values of γ were evaluated: 1.0, 1.3, and 1.5. Results in Table 3.16 show only minor differences across these settings, especially for γ values 1.3 and 1.5.

However, as these results were obtained on the orthophoto dataset, I conducted an additional validation experiment on Sentinel-2 data to account for potential sensitivity

to sensor characteristics. To avoid favouring any SR method, this test was performed on bilinearly resampled Sentinel-2 tiles. Again, the same γ values were tested as before. Table 3.17 demonstrates that $\gamma = 1.3$ achieves the best results in this setting. This can be attributed to the lower spatial resolution of Sentinel-2 images, where a moderately stronger focus on difficult pixels is beneficial, but excessive weighting (with $\gamma = 1.5$), as suggested by the experiment on orthophoto images, seems to overcompensate and leads to lower metric scores.

Table 3.16: Models trained on the orthophotos with different FTL parameters to determine which one performs best. The best results on the metrics IoU and F_1 are highlighted in bold.

Name	**Loss Function and γ Value**	**Sensor**	**Precision**	**Recall**	**IoU**	F_1
10	FTL with $\gamma = 1.5$	Orthophoto	0.768	0.638	**0.556**	**0.684**
02	FTL with $\gamma = 1.3$	Orthophoto	0.769	0.633	0.552	0.681
09	FTL with $\gamma = 1.0$	Orthophoto	0.776	0.624	0.549	0.679

Table 3.17: Models trained on the bilinearly interpolated Sentinel-2 images with different FTL parameters to determine which one performs best. The best results on the metrics IoU and F_1 are highlighted in bold.

Name	**Loss Function and γ Value**	**Sensor**	**Precision**	**Recall**	**IoU**	F_1
12	FTL with $\gamma = 1.3$	Sentinel-2 RGBN	0.532	0.464	**0.352**	**0.481**
13	FTL with $\gamma = 1.5$	Sentinel-2 RGBN	0.562	0.422	0.339	0.467
11	FTL with $\gamma = 1.0$	Sentinel-2 RGBN	0.558	0.423	0.339	0.466

The results of all trained models during these experiments are presented in Table 3.18. It provides an overview of different setup parameters and facilitates the comparison of selected model configurations.

Table 3.18: Test results for all models used during the experiments to determine setup parameters.

Name	**Precision**	**Recall**	**IoU**	F_1
01	0.766	0.611	0.538	0.666
02	0.769	0.633	0.552	0.681
03	0.756	0.610	0.535	0.661
04	0.766	0.614	0.539	0.668
04_1	0.723	0.582	0.502	0.633
04_2	0.733	0.575	0.501	0.632
05	0.818	0.497	0.463	0.601
06	0.783	0.597	0.531	0.664
07	0.723	0.572	0.499	0.627
08	0.766	0.627	0.546	0.676
09	0.776	0.624	0.549	0.679
10	0.768	0.638	0.556	0.684
11	0.558	0.423	0.339	0.466
12	0.532	0.464	0.352	0.481
13	0.562	0.422	0.339	0.467

3.4.5.4 Training on Super-Resolved Images

Based on the presented findings all building delineation models are trained with the following configuration:

- Dataset: **Dataset Filtered**
- Model architecture: UNet with se_ResNext50 backbone
- Loss function: FTL with $\alpha = 0.7$, $\beta = 0.3$, $\gamma = 1.3$
- Optimiser: Adam
- Pre-training: From scratch
- Learning Rate Scheduler:
  - Patience: 5
  - Factor: 0.5
- Early Stopping:
  - Patience: 10

With the training parameters established, the final step is training the individual models on the super-resolved datasets. This required defining configuration files for each SR model to adjust data storage paths and file names. After training, the weights from the best-performing epoch were selected and applied to the respective test dataset to determine the model's performance and provide sample images representing the model's building delineation capabilities. All results were logged via the *wandb* API, enabling efficient tracking and comparison of models.

CHAPTER 4

Results

This chapter presents the experimental results obtained from applying the SR and building delineation models. The findings provide a basis for assessing how different SR approaches influence delineation performance and contribute to answering the RQs in Chapter 5.

4.1 Image Samples

The images provided here set the foundation for a qualitative assessment of each model's performance using a random sample from the test dataset. The sampling was performed consistently across all models to ensure a fair comparison. Additional samples and full resolution images can be accessed in the accompanying repository provided in Chapter 1. All Sentinel-2 images are shown using only their RGB bands, whereas the building delineation outputs are presented as single-band binary images.

4.1.1 Super-Resolution Examples

First, a comparison of the SR results from each model on randomly selected images is provided in Figure 4.1. Each row displays the interpolated or super-resolved images for three distinct scenes covering a rural (Jenks stratification class 1 or 2), rural-urban (Jenks stratification class 3 or 4) and an urban area (Jenks stratification class 5).

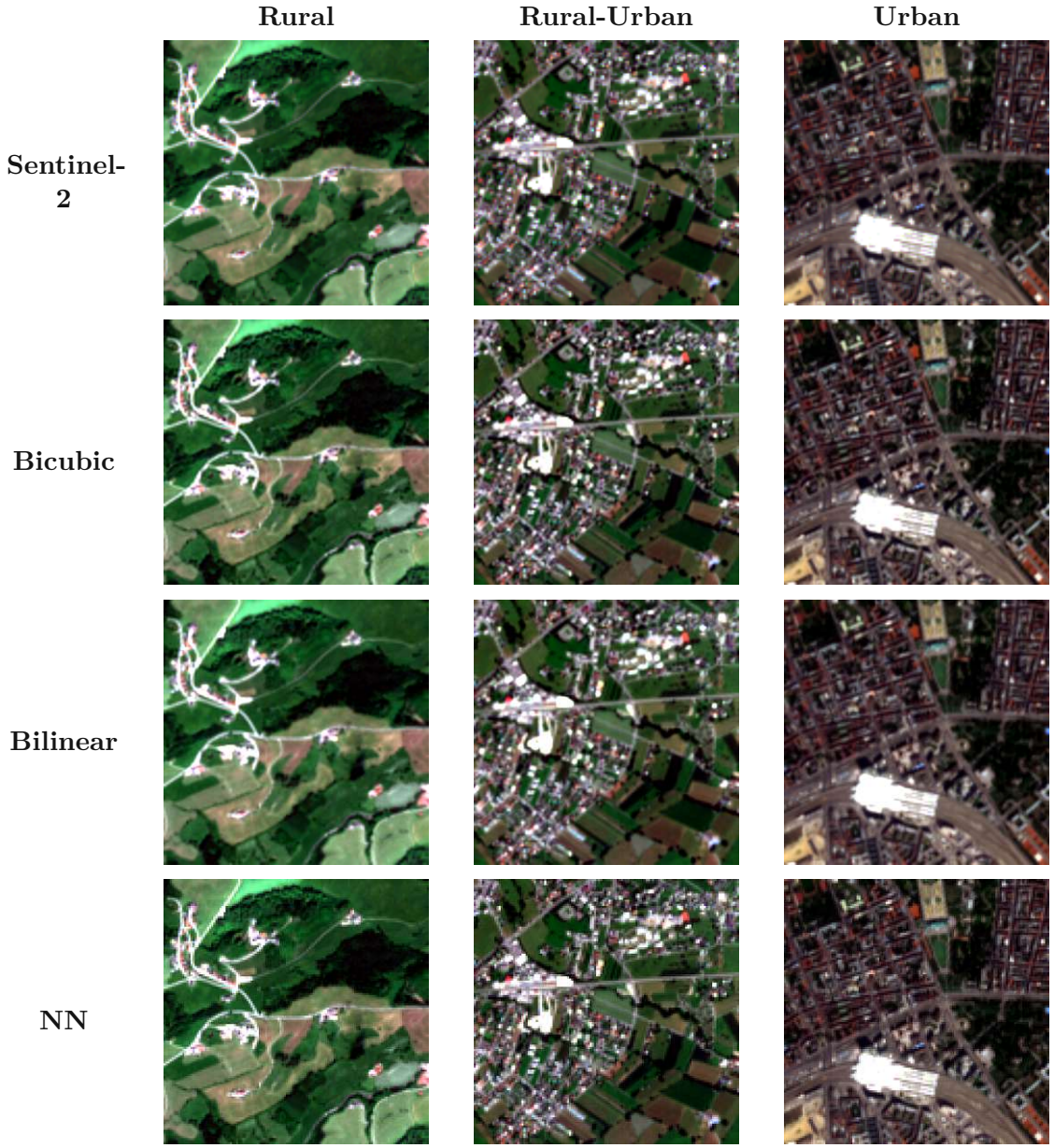


Figure 4.0: Sample of interpolated and super-resolved images from all models for three areas (rural, rural-urban, and urban). Each image is of shape $4 \times 512 \times 512$ pixels which covers an area of 1280×1280 m^2.

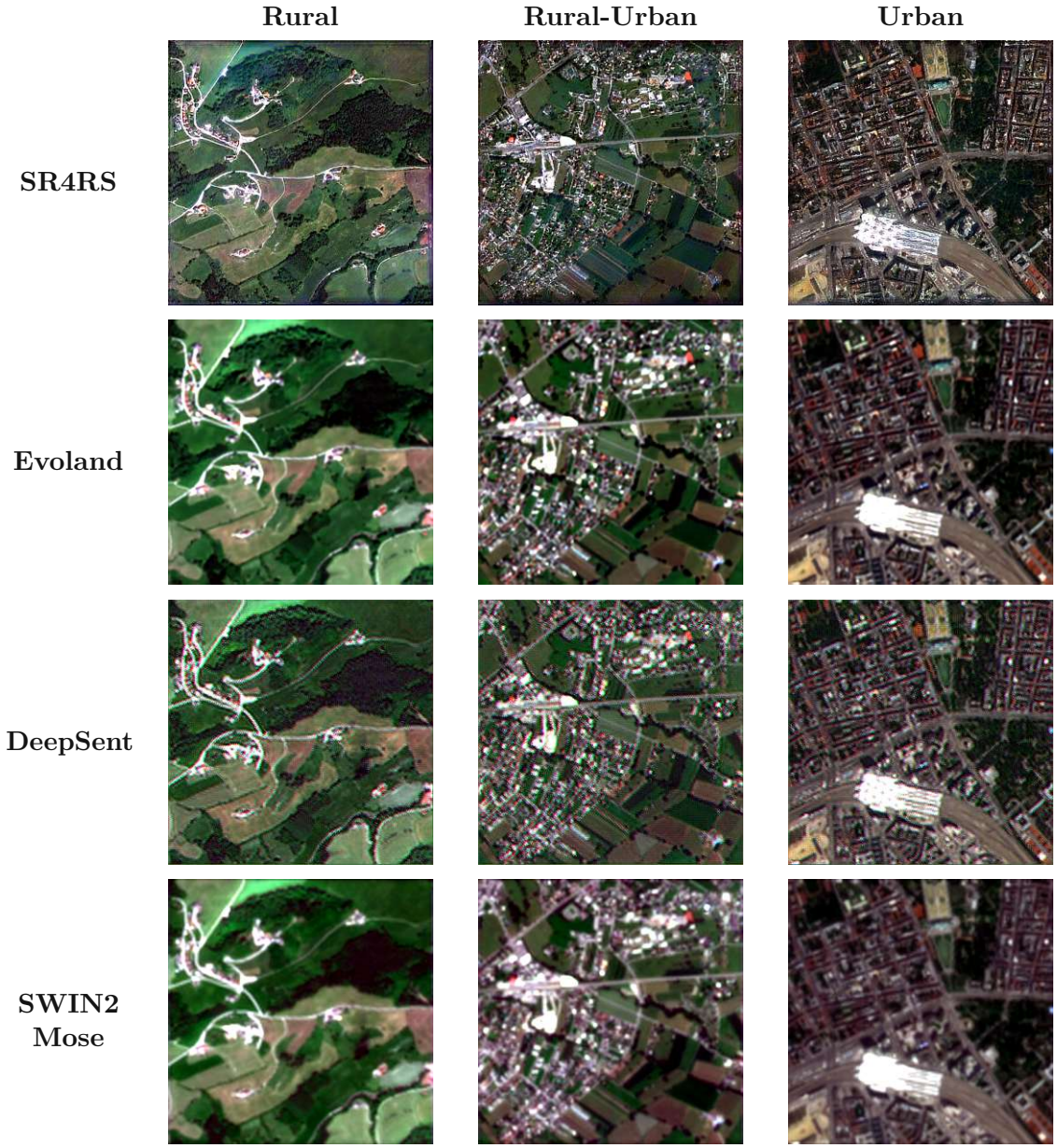


Figure 4.0: Sample of interpolated and super-resolved images from all models for three areas (rural, rural-urban, and urban). Each image is of shape $4 \times 512 \times 512$ pixels which covers an area of 1280×1280 m^2.

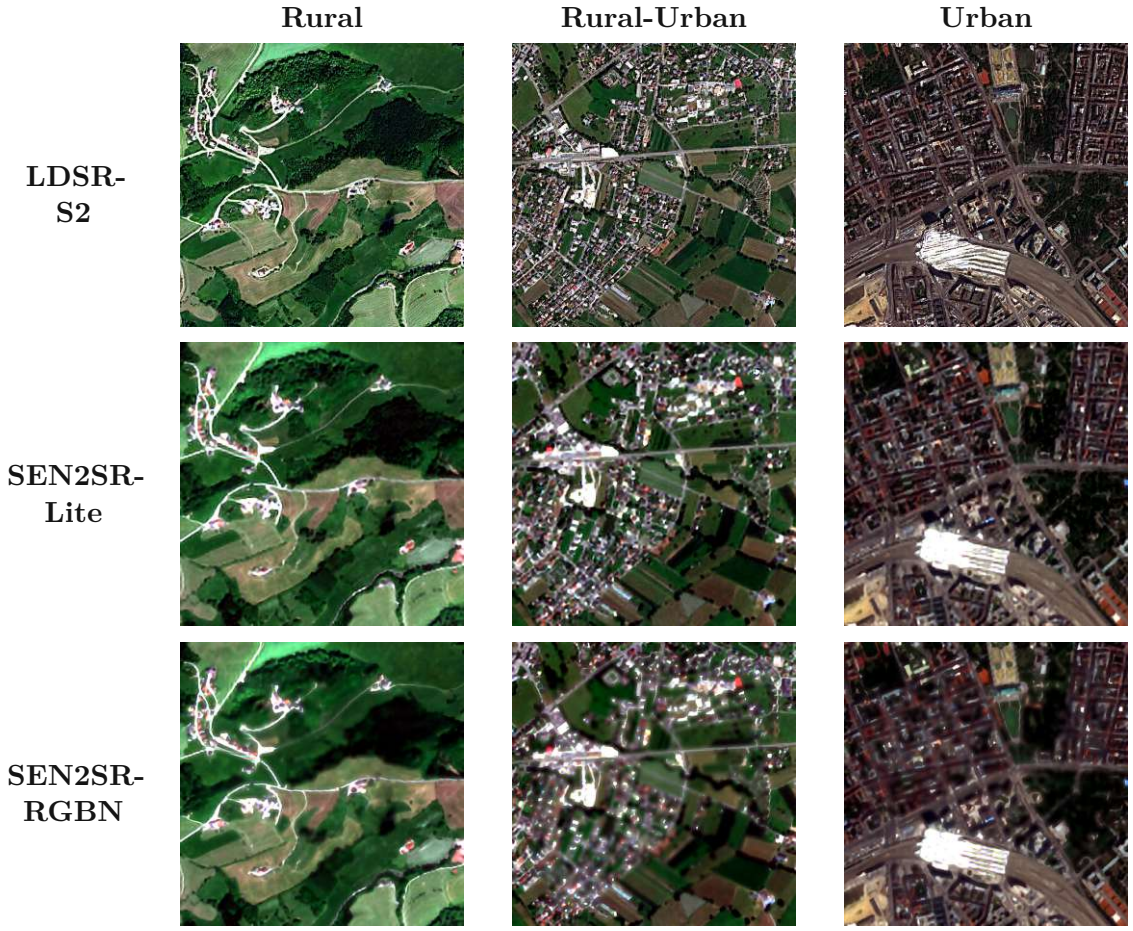


Figure 4.0: Sample of interpolated and super-resolved images from all models for three areas (rural, rural-urban, and urban). Each image is of shape $4 \times 512 \times 512$ pixels which covers an area of 1280×1280 m^2.

While a qualitative assessment alone cannot provide a conclusive evaluation of different interpolation and SR methods, it supports the interpretation of quantitative, metric-based results. Visual inspection of imagery can reveal patterns, distortions, and spectral shifts that may not be captured by numerical metrics. Although subjective judgment from human observers introduces potential bias, such qualitative analysis complements quantitative evaluation and offers additional insights into model performance.

As described in Section 2.1, the interpolation techniques do not introduce any high-frequency information. As expected, the **NN** interpolation looks identical to the original Sentinel-2 image but with a higher spatial resolution. Both the **bilinear** and the **bicubic**

interpolations visually smooth the input, but discernible differences between them could not be determined.

Compared to the interpolation methods, the application of SR models introduces HR information and can possibly improve building delineation results. The outputs of **SR4RS** generally look super-resolved, but hallucinations are noticeable. Urban areas in particular appear somewhat unrealistic, with irregular shapes and non-straight edges. The ringing border effects due to buffering during inference are noticeable, but do not lead to a visible deterioration of the image. **Evoland** produces results that are closer in appearance to Sentinel-2 than to other SR outputs. This is likely due to the SR to 5 m followed by bilinear interpolation to 2.5 m, which retains Sentinel-2's image features more closely. Spectral properties seem well aligned, and no clear hallucinations were observed. The images from **DeepSent** have a strange appearance, with a visible noise-like pattern across all samples. Aside from that, spatial structures seem to be preserved reasonably well. **Swin2Mose** appears very similar to **Evoland**, with no obvious differences visible in the samples checked manually. **LDSR-S2** is somewhat comparable to **SR4RS** but with fewer apparent hallucinations. Rural areas look convincing, though in urban scenes, certain structures, like roads and paths in parks, appear less consistent. **SEN2SR-Lite** gives the impression of being smoothed. Shadows and dark parts of super-resolved images stand out strongly, while straight features appear well reconstructed. Overall, it does not quite resemble imagery at 2.5 m resolution to a human observer. Finally, **SEN2SR-RGBN** appears even more smoothed than **SEN2SR-Lite**, with homogenous areas appearing heavily blurred (e.g., forests and meadows), while at the same time, boundaries are distinctly reconstructed.

4.1.2 Building Delineation Examples

Figures 4.1 to 4.10 illustrate the outputs of each building delineation model. In each figure, the rows are organised as follows: each one corresponds to a single image sample from one of the Jenks-derived stratification classes, reflecting its building pixel percentage. The displayed images are $4 \times 256 \times 256$ subsamples extracted from the original $4 \times 512 \times 512$ input image, rather than the full input image, as these are the prediction maps produced by each delineation model. The first column shows the interpolated or super-resolved input image fed into the building delineation model, the second column displays the GT building footprints, the third column presents the predicted binary building footprint map, and the fourth column shows the pixel confusion map (Pixel-wise Confusion Map (PCM)). In the PCM, each pixel is colour-coded according to its classification: True Negative (TN) (correct background) in black, TP (correctly identified buildings) in green, FP (background pixels incorrectly predicted as buildings) in red, and FN (missed building pixels) in blue. This visualisation helps identify whether a model tends to over- or under-predict, and which types of buildings it handles well or struggles with.

The building delineation results for different interpolation and SR methods show only minor variations overall, yet notable differences between approaches can still be observed. The visual inspection of these results is facilitated by the corresponding colourised PCM, which highlights correctly detected and missed buildings. However, it must be emphasized that the very limited sample size, one image per Jenks-derived class, cannot be considered representative. Therefore, the following observations illustrate trends visible in the provided samples rather than statistically significant insights. Across all samples, models generally struggled with similar image regions while performing consistently well in others.

For **class 0**, containing no buildings, most models correctly identified the absence of structures. Nonetheless, several methods, such as NN, SR4RS, Evoland, DeepSent, and LDSR-S2, produced FP, generating non-existent building artefacts. For **class 1**, which includes a few larger structures in the upper-left part of the image, all models successfully detected the main buildings. Smaller structures located below them were only partially recognised, and some very small buildings in the lower-right corner were inconsistently labelled. Only the network trained on SR4RS imagery failed to detect these very small buildings entirely. With the increased building density in **class 2**, visual evaluation becomes more challenging. Most models correctly delineated major structures near the image centre but frequently missed smaller, dispersed buildings. Larger or clustered buildings were generally detected, though all models tended to underpredict the amount of building pixels. No clear performance difference between interpolation and SR-based inputs could be discerned. The **class 3** sample, containing numerous larger and clustered buildings, revealed a tendency among all models to overpredict building extents, particularly in the large agglomerated complex in the upper-left corner. This effect possibly results from spectral similarities between buildings and adjacent surfaces, which complicates the building delineation tremendously. Overall, the models performed comparably, with no consistent outlier detected. For **class 4**, featuring a large continuous building area and several isolated blocks, all methods struggled to delineate the continuous building accurately. Differences were mainly observed in the extent of the area labelled as buildings and in the detection of separated building complexes. Interpolation-based methods and certain SR approaches, notably NN, bicubic, SR4RS, and LDSR-S2, tended to underpredict in the upper-right and lower-left corners, while other models, such as DeepSent, over- and underpredicted across the image. Finally, in **class 5**, which contains densely built-up urban areas, all models achieved visually convincing results. Although minor misclassifications remain at building edges and within courtyards, the overall shape and structure of the buildings were accurately captured across all methods.

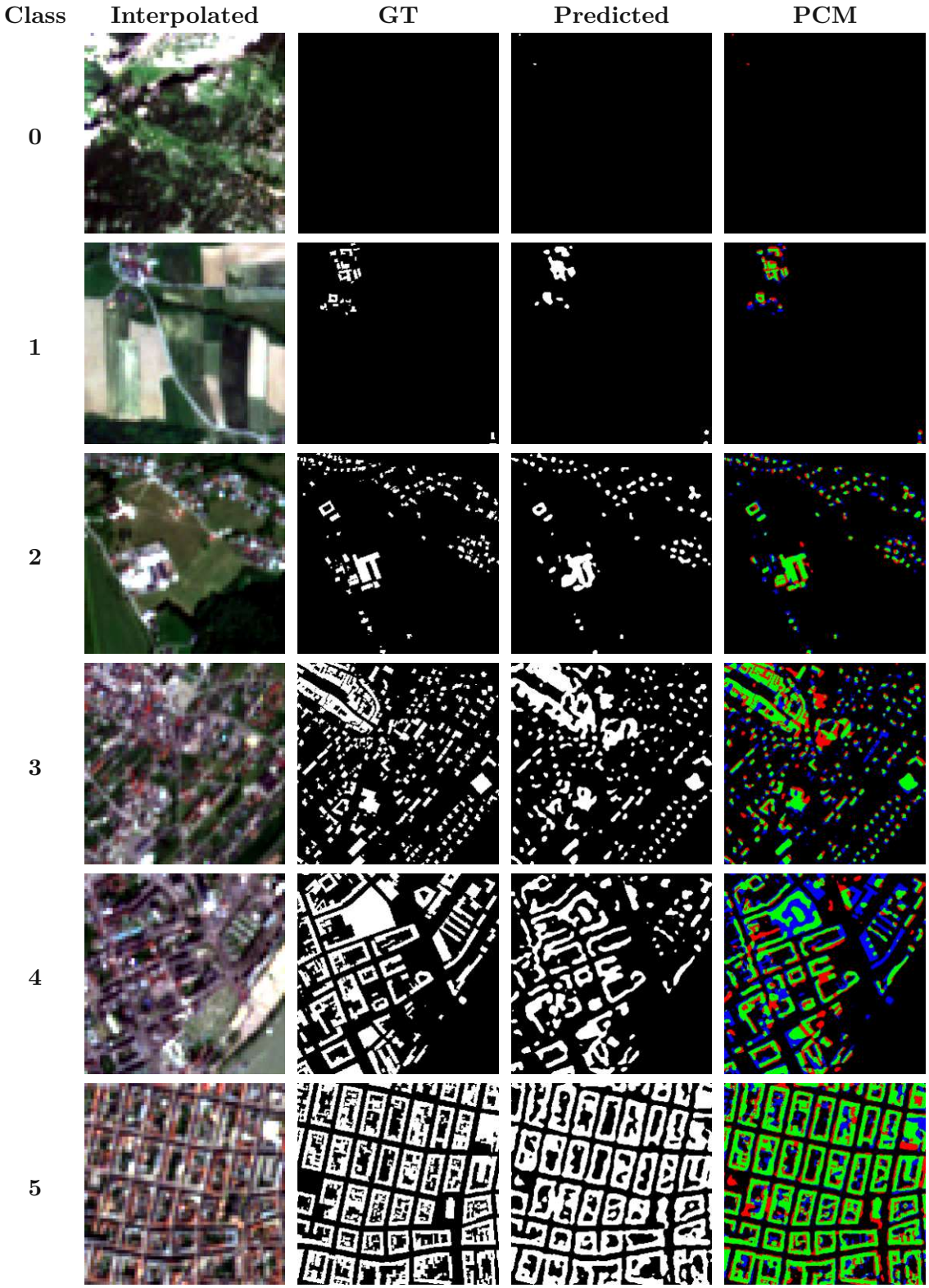


Figure 4.1: **NN** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

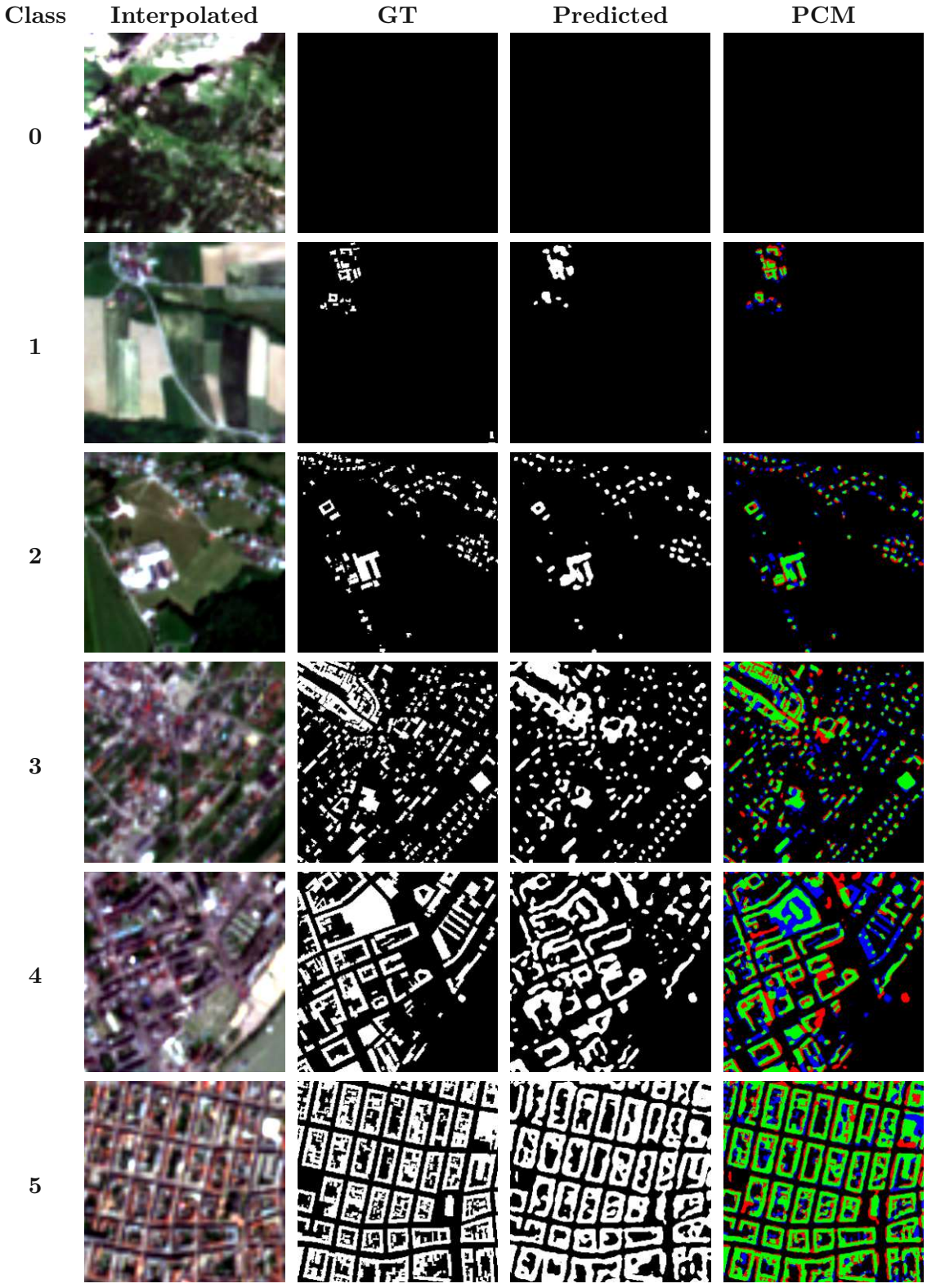


Figure 4.2: **Bilinear** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

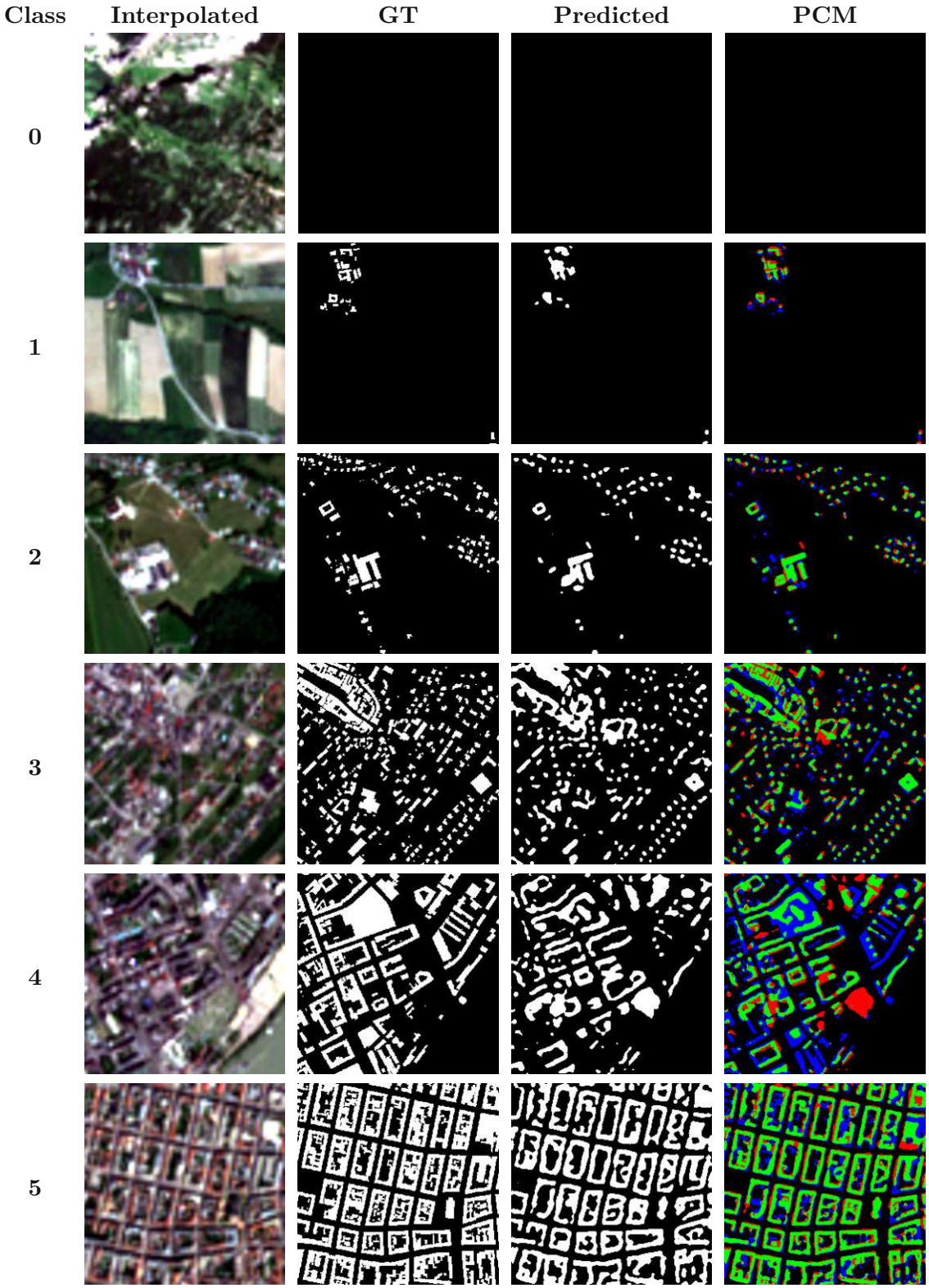


Figure 4.3: **Bicubic** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

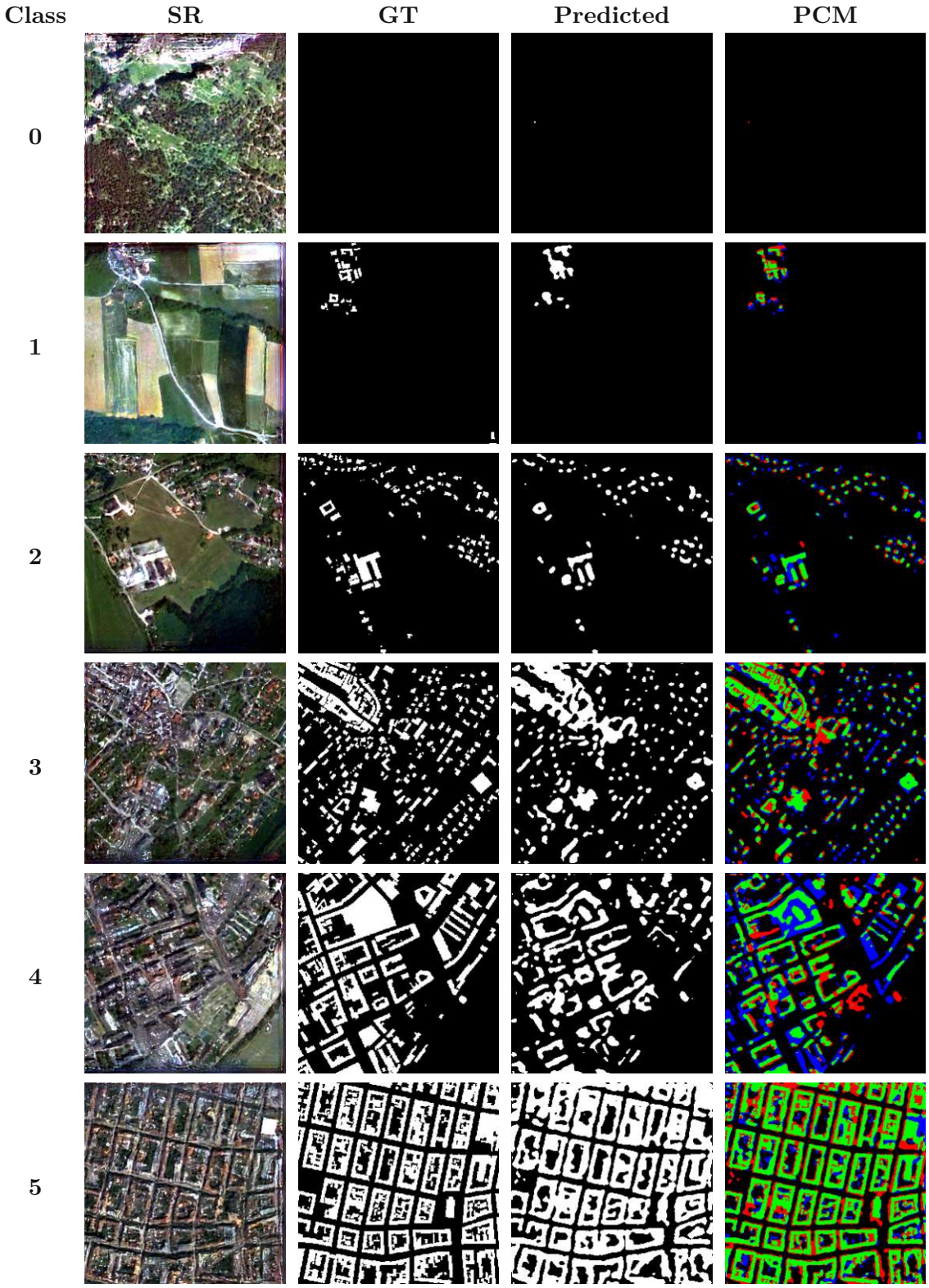


Figure 4.4: **SR4RS** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

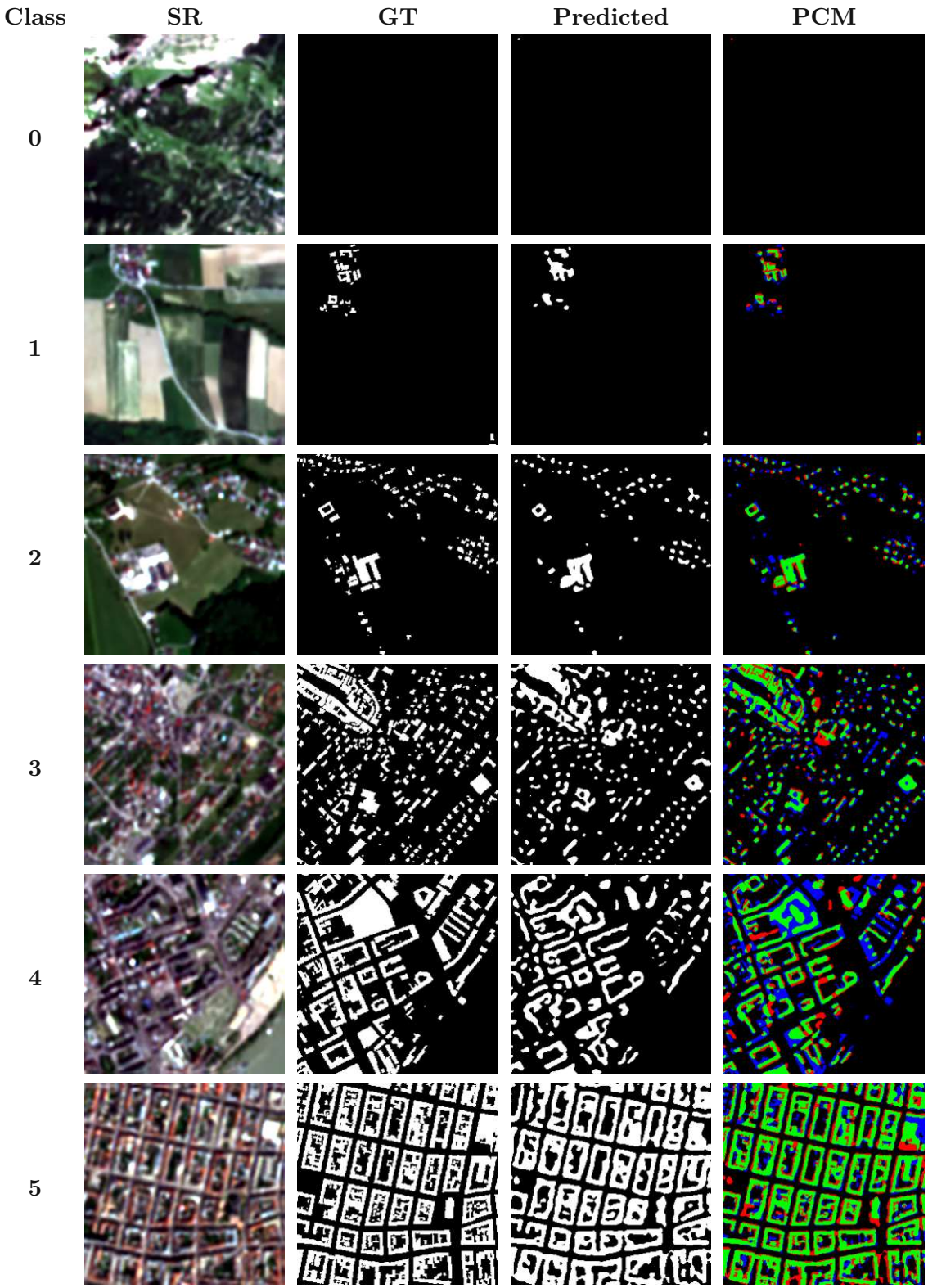


Figure 4.5: **Evoland** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

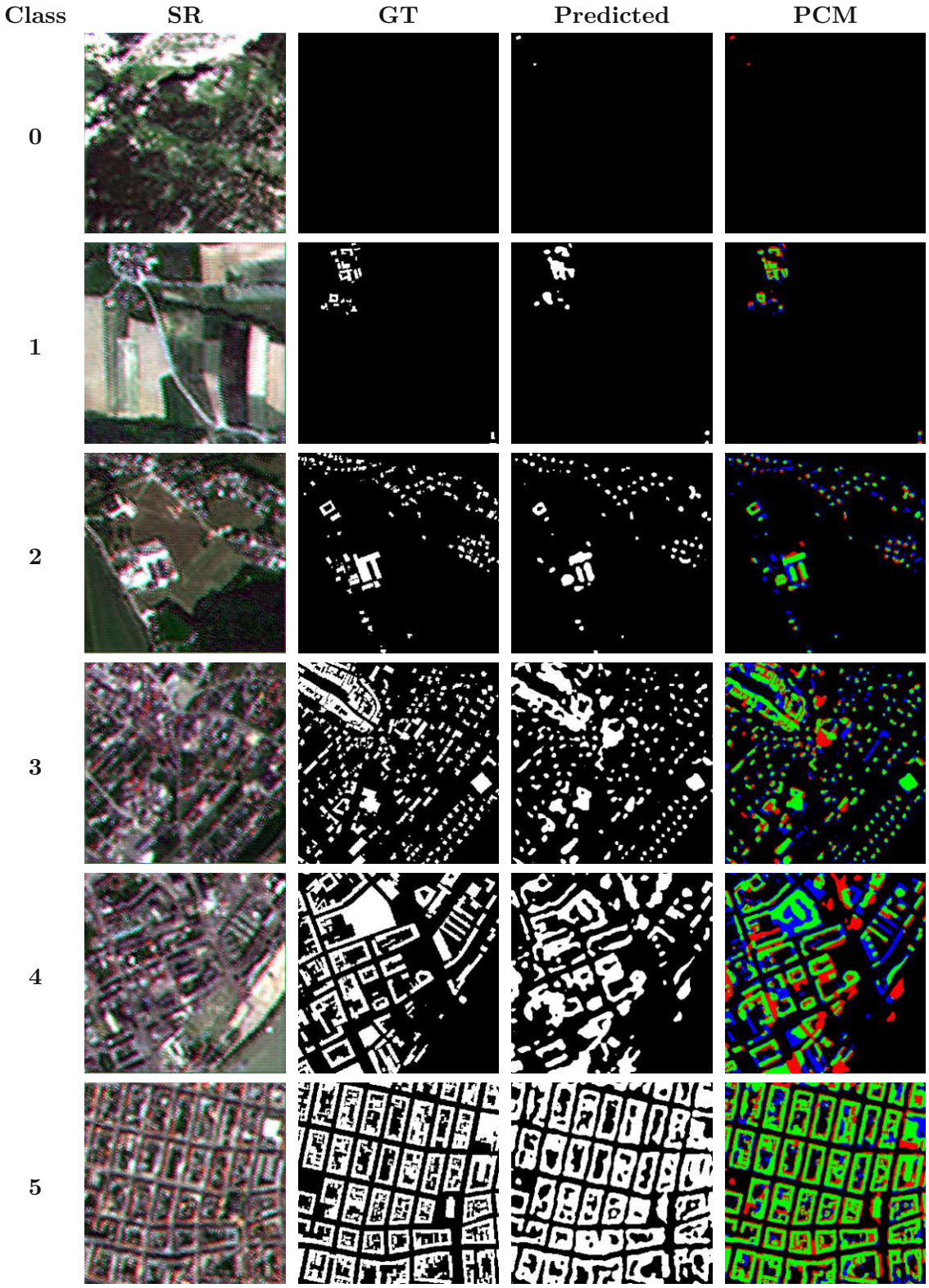


Figure 4.6: **DeepSent** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

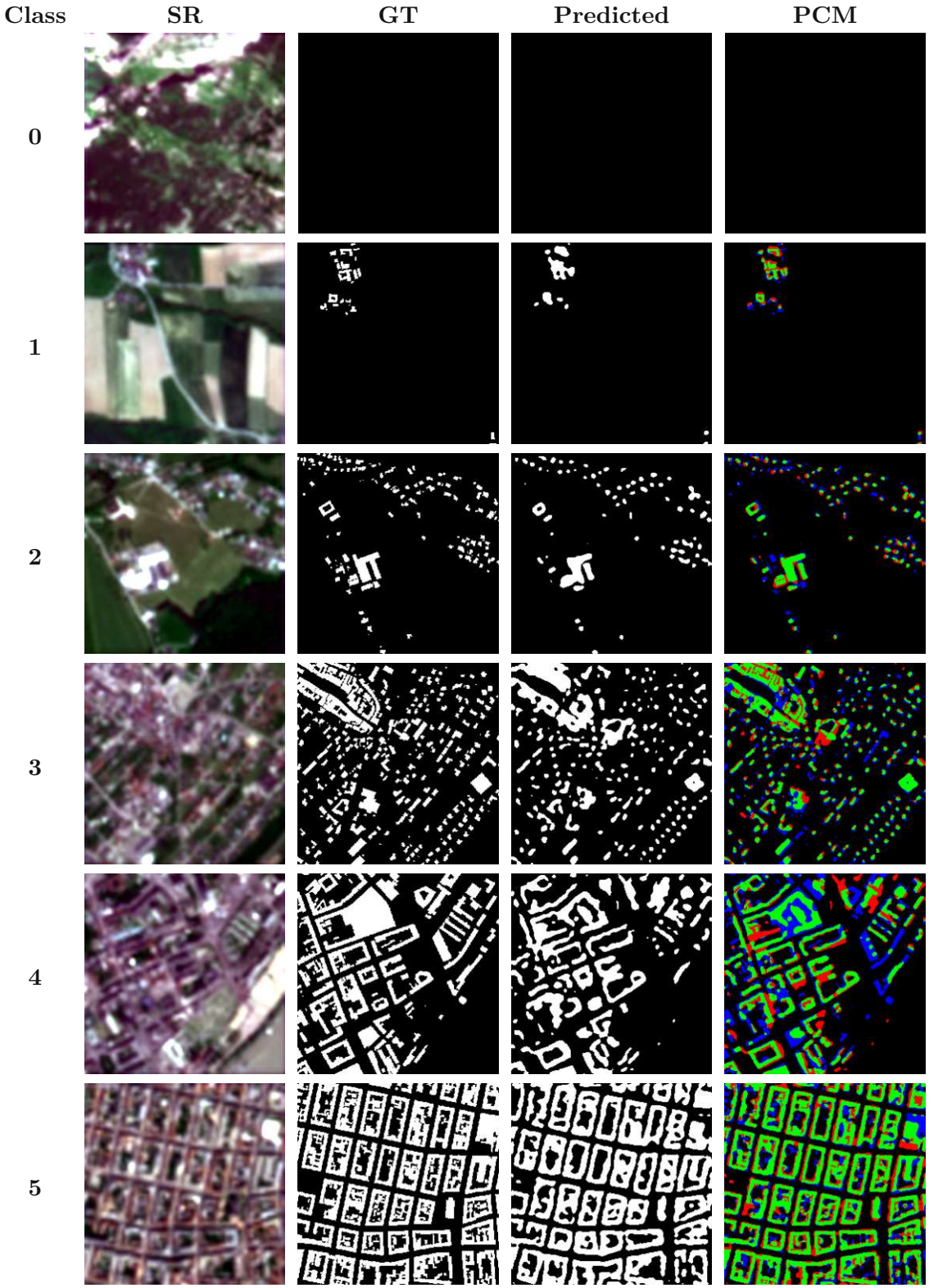


Figure 4.7: **SWIN2Mose** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

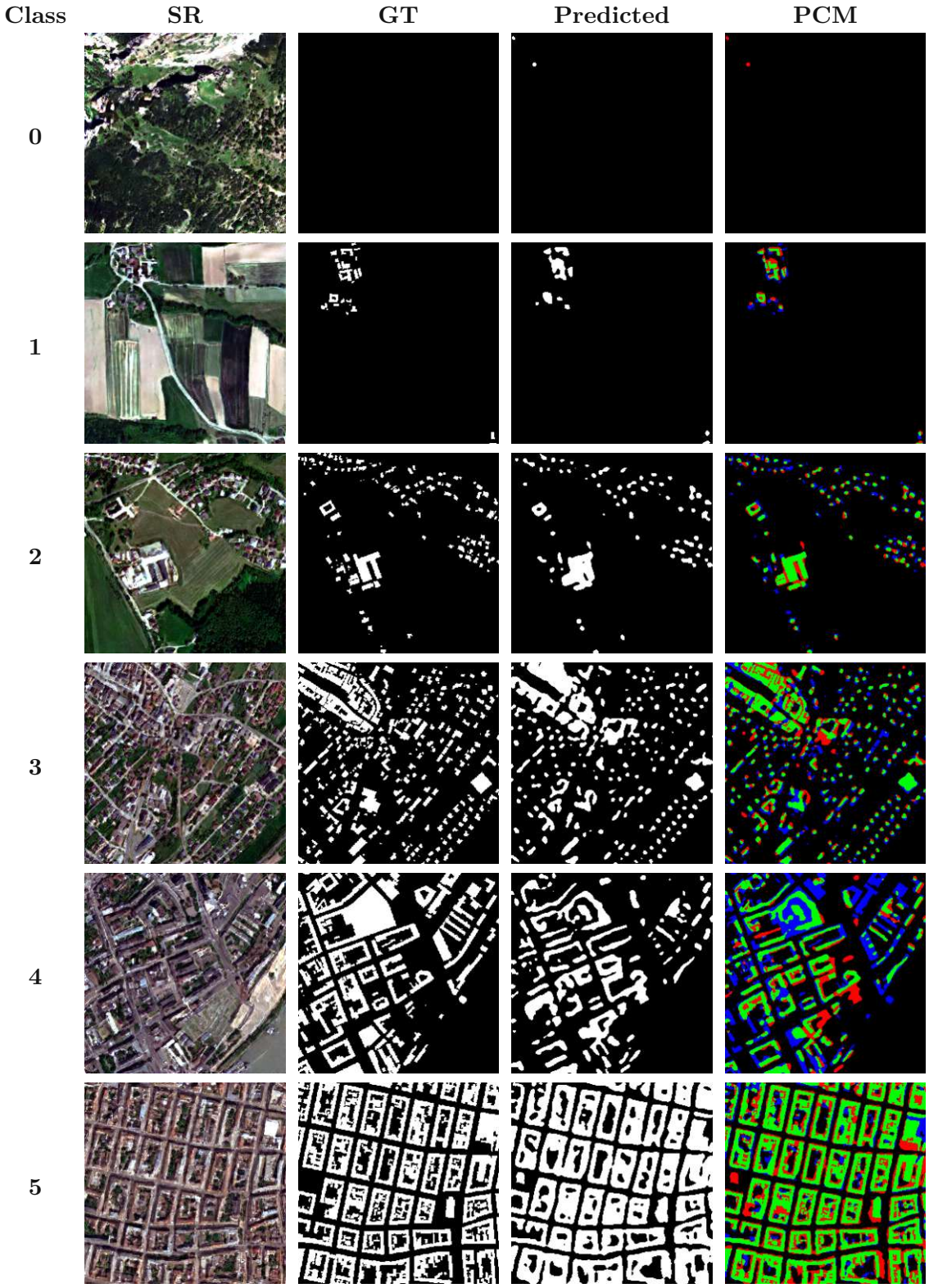


Figure 4.8: **LDSR-S2** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

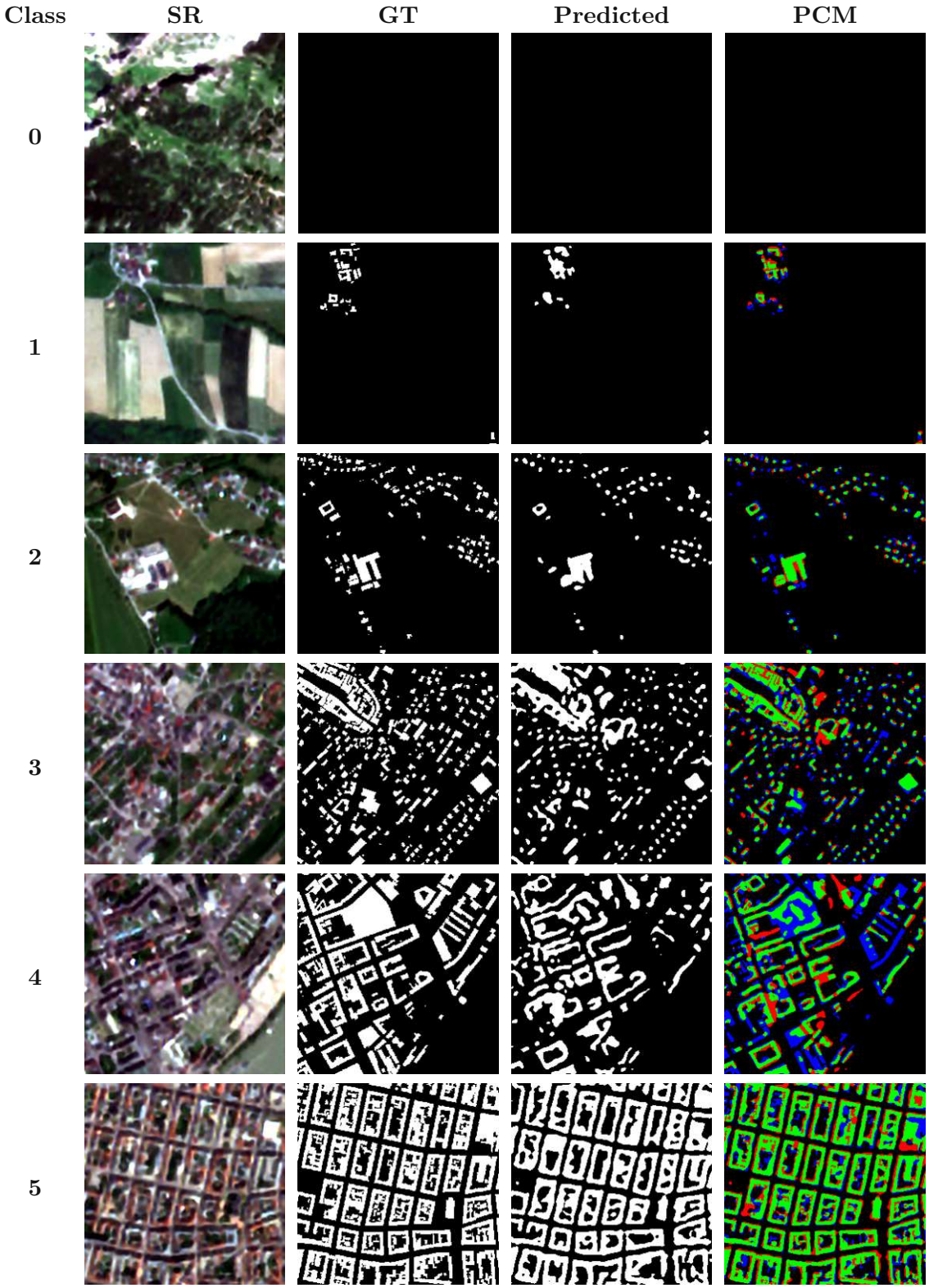


Figure 4.9: **SEN2SR-Lite** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

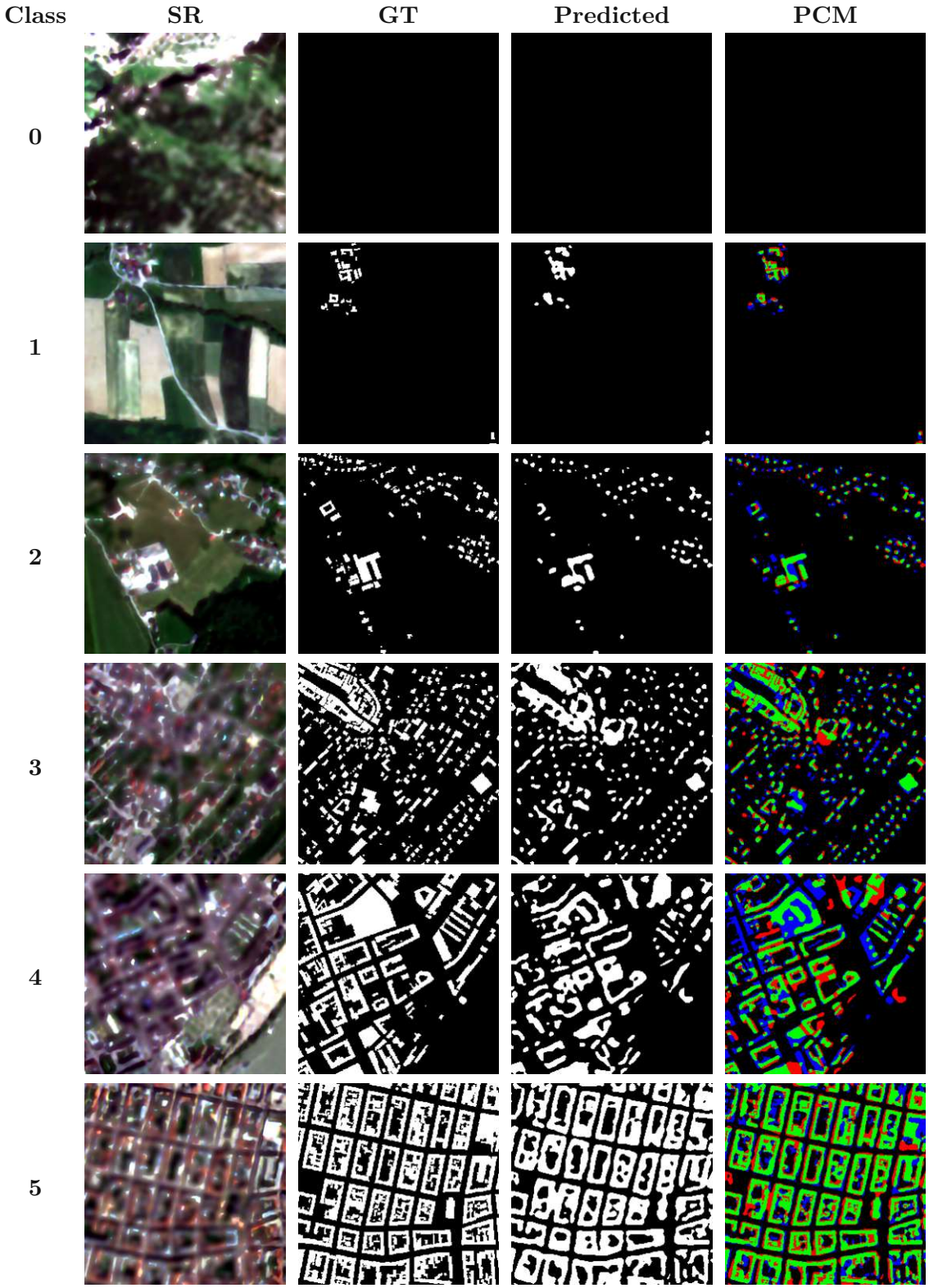


Figure 4.10: **SEN2SR-RGBN** sample images, with each image of shape $1 \times 256 \times 256$ pixels which covers an area of 640×640 m^2. PCM color-coding: TN black; TP green; FP red; FN blue.

4.2 Building Delineation Results

The building delineation model performance for all datasets is summarized in Table 4.2, using the best-performing set of model weights based on validation results. To further assess generalization across different environments, metrics were also computed for the Jenks-derived stratification classes presented in Tables 4.3 to 4.8. The object-based metrics (OPA and OFA) are reported in Tables 4.9 and 4.10. These yet untested metrics provide additional insights into building-level prediction quality, both averaged over all images and grouped by building size.

The results presented in Table 4.2 show clear performance differences between the evaluated methods. Building delineation models trained on orthophoto images achieved the highest scores with IoU= 0.552 and $F_1 = 0.681$, outperforming all other approaches by approximately 20%. Among the interpolation techniques, NN interpolation yielded the best results with IoU= 0.354 and $F_1 = 0.485$, while bilinear and bicubic interpolations followed closely within a 1% margin. The best-performing SR model, SEN2SR-RGBN, reached IoU= 0.349 and $F_1 = 0.478$, followed by Evoland, SWIN2Mose, and SEN2SR-Lite, all within 1% of these scores. SR4RS, LDSR-S2, and DeepSent performed moderately worse, with scores up to 3.3% below SEN2SR-RGBN in F_1. This establishes a clear order of performance for all methods: orthophoto-based models performed best, followed by interpolation, and then SR-based models.

Table 4.2: Comparison of models on the building segmentation task. The best results for each category for IoU and F_1 are highlighted in bold. Models are sorted by performance in F_1.

Name	Precision	Recall	IoU	F_1
Orthophoto Baseline	0.769	0.633	**0.552**	**0.681**
Nearest Neigbour	0.545	0.459	**0.354**	**0.485**
Bilinear	0.532	0.464	0.352	0.481
Bicubic	0.545	0.446	0.348	0.477
SEN2SR-RGBN	0.563	0.438	**0.349**	**0.478**
Evoland	0.564	0.431	0.346	0.475
SWIN2Mose	0.552	0.436	0.344	0.473
SEN2SR-Lite	0.559	0.427	0.341	0.468
SR4RS	0.533	0.416	0.325	0.451
LDSR-S2	0.528	0.416	0.323	0.450
DeepSent	0.529	0.409	0.320	0.445

Analysis driven by Jenks-derived stratification classes featured in Tables 4.3 to 4.8 provides additional insight into model behaviour across different levels of urbanization. The overall ranking of methods remains mostly consistent with the aggregated results presented above. A notable exception are the results for class 0, featuring images without buildings, where SR models perform best, followed by interpolation methods and then the orthophoto baseline. Among the interpolation methods, bilinear interpolation achieved the highest IoU in classes 2 and 4 and the highest F_1 in class 5, while NN interpolation performed best in classes 1, 3, and 4 for F_1. Bicubic interpolation achieved the top result in class 0 (non-building areas) and for IoU in class 5. Among the SR models, SEN2SR-RGBN consistently outperformed the others across most stratification classes. Notable exceptions include class 0, where Evoland achieved the best overall result, and class 4, where Evoland slightly surpassed SEN2SR-RGBN in IoU.

Table 4.3: Comparison of models on class 0 with 1,133 images. The best results for each category for IoU and F_1 are highlighted in bold. Models are sorted in the same order as in Table 4.2.

Name	**Precision**	**Recall**	**Specificity**	**IoU**	F_1
Orthophoto	0.888	0.888	1.000	**0.888**	**0.888**
Nearest Neighbour	0.900	0.900	1.000	0.900	0.900
Bilinear	0.894	0.894	1.000	0.894	0.894
Bicubic	0.918	0.918	1.000	**0.918**	**0.918**
SEN2SR-RGBN	0.941	0.941	1.000	0.941	0.941
Evoland	0.971	0.971	1.000	**0.971**	**0.971**
Swin2Mose	0.953	0.953	1.000	0.953	0.953
SEN2SR-Lite	0.953	0.953	1.000	0.953	0.953
SR4RS	0.941	0.941	0.999	0.941	0.941
LDSR-S2	0.912	0.912	1.000	0.912	0.912
DeepSent	0.900	0.900	1.000	0.900	0.900

Table 4.4: Comparison of models on class 1 with 16,606 images. The best results for each category for IoU and F_1 are highlighted in bold. Models are sorted in the same order as in Table 4.2.

Name	**Precision**	**Recall**	**Specificity**	**IoU**	F_1
Orthophoto	0.748	0.599	0.9980	**0.517**	**0.650**
Nearest Neighbour	0.503	0.406	0.9960	**0.303**	**0.432**
Bilinear	0.486	0.410	0.9957	0.300	0.427
Bicubic	0.498	0.393	0.9962	0.296	0.423
SEN2SR-RGBN	0.519	0.382	0.9967	**0.296**	**0.423**
Evoland	0.516	0.374	0.9966	0.291	0.418
Swin2Mose	0.505	0.379	0.9965	0.290	0.417
SEN2SR-Lite	0.511	0.379	0.9965	0.294	0.421
SR4RS	0.486	0.357	0.9964	0.270	0.392
LDSR-S2	0.484	0.360	0.9964	0.271	0.396
DeepSent	0.484	0.356	0.9965	0.269	0.392

Table 4.5: Comparison of models on class 2 with 3,568 images. The best results for each category for IoU and F_1 are highlighted in bold. Models are sorted in the same order as in Table 4.2.

Name	**Precision**	**Recall**	**Specificity**	**IoU**	F_1
Orthophoto	0.808	0.685	0.9984	**0.590**	**0.738**
Nearest Neighbour	0.598	0.533	0.9963	0.393	0.558
Bilinear	0.594	0.545	0.9961	**0.398**	**0.563**
Bicubic	0.607	0.521	0.9965	0.390	0.555
SEN2SR-RGBN	0.619	0.502	0.9971	**0.385**	**0.548**
Evoland	0.626	0.491	0.9970	0.380	0.544
Swin2Mose	0.611	0.500	0.9969	0.380	0.544
SEN2SR-Lite	0.611	0.495	0.9970	0.381	0.542
SR4RS	0.594	0.486	0.9968	0.364	0.528
LDSR-S2	0.581	0.479	0.9968	0.356	0.518
DeepSent	0.595	0.462	0.9969	0.350	0.511

Table 4.6: Comparison of models on class 3 with 964 images. The best results for each category for IoU and F_1 are highlighted in bold. Models are sorted in the same order as in Table 4.2.

Name	**Precision**	**Recall**	**Specificity**	**IoU**	F_1
Orthophoto	0.813	0.690	0.9984	**0.595**	**0.743**
Nearest Neighbour	0.611	0.543	0.9963	**0.404**	**0.571**
Bilinear	0.607	0.543	0.9963	0.402	0.570
Bicubic	0.626	0.508	0.9966	0.390	0.556
SEN2SR-RGBN	0.625	0.514	0.9969	**0.395**	**0.562**
Evoland	0.633	0.503	0.9968	0.390	0.557
Swin2Mose	0.616	0.512	0.9968	0.389	0.556
SEN2SR-Lite	0.616	0.507	0.9968	0.389	0.556
SR4RS	0.601	0.491	0.9967	0.369	0.535
LDSR-S2	0.593	0.505	0.9967	0.374	0.539
DeepSent	0.593	0.493	0.9967	0.367	0.533

Table 4.7: Comparison of models on class 4 with 348 images. The best results for each category for IoU and F_1 are highlighted in bold. Models are sorted in the same order as in Table 4.2.

Name	**Precision**	**Recall**	**Specificity**	**IoU**	F_1
Orthophoto	0.829	0.702	0.9985	**0.614**	**0.760**
Nearest Neighbour	0.668	0.561	0.9964	0.421	**0.604**
Bilinear	0.649	0.571	0.9964	**0.435**	0.603
Bicubic	0.683	0.520	0.9968	0.419	0.587
SEN2SR-RGBN	0.673	0.550	0.9968	0.414	**0.604**
Swin2Mose	0.659	0.540	0.9967	0.406	0.594
Evoland	0.677	0.544	0.9968	**0.431**	0.598
SEN2SR-Lite	0.665	0.545	0.9968	0.409	0.596
SR4RS	0.639	0.529	0.9966	0.398	0.572
LDSR-S2	0.632	0.533	0.9967	0.402	0.578
DeepSent	0.626	0.520	0.9967	0.393	0.561

Table 4.8: Comparison of models on class 5 with 46 images. The best results for each category for IoU and F_1 are highlighted in bold. Models are sorted in the same order as in Table 4.2.

Name	**Precision**	**Recall**	**Specificity**	**IoU**	F_1
Orthophoto	0.825	0.710	0.860	**0.602**	**0.763**
Nearest Neighbour	0.739	0.608	0.833	0.488	0.665
Bilinear	0.736	0.615	0.835	0.492	**0.670**
Bicubic	0.748	0.604	0.831	**0.494**	0.655
SEN2SR-RGBN	0.744	0.606	0.834	**0.491**	**0.664**
Evoland	0.741	0.601	0.832	0.489	0.660
Swin2Mose	0.734	0.602	0.831	0.484	0.660
SEN2SR-Lite	0.739	0.599	0.832	0.486	0.658
SR4RS	0.728	0.593	0.829	0.480	0.653
LDSR-S2	0.730	0.595	0.829	0.481	0.655
DeepSent	0.725	0.590	0.828	0.476	0.650

The analysis of the OPA and OFA metrics, shown in Tables 4.9 and 4.10, aligns with the previous findings. Models trained on orthophoto imagery achieved the highest scores, followed by interpolation and SR-based models. Among the interpolation techniques, bilinear interpolation performed best, closely followed by NN. Among the SR models SEN2SR-RGBN ranked highest, with SWIN2Mose and Evoland producing results within 1% of its scores. A similar ordering of performances was observed when metrics were sorted and analysed by object size.

Table 4.9: Comparison of object-level prediction averages and overall found fraction across models. The best results for each category for OPA and OFA are highlighted in bold. Models are sorted by performance in OFA.

Name	OPA	OFA
Orthophoto Baseline	**0.476**	**0.581**
Bilinear	**0.285**	**0.322**
Nearest Neighbour	0.284	0.319
Bicubic	0.276	0.310
SEN2SR-RGBN	**0.267**	**0.292**
SWIN2Mose	0.262	0.291
Evoland	0.260	0.288
SEN2SR-Lite	0.255	0.278
LDSR-S2	0.253	0.272
SR4RS	0.248	0.270
DeepSent	0.245	0.265

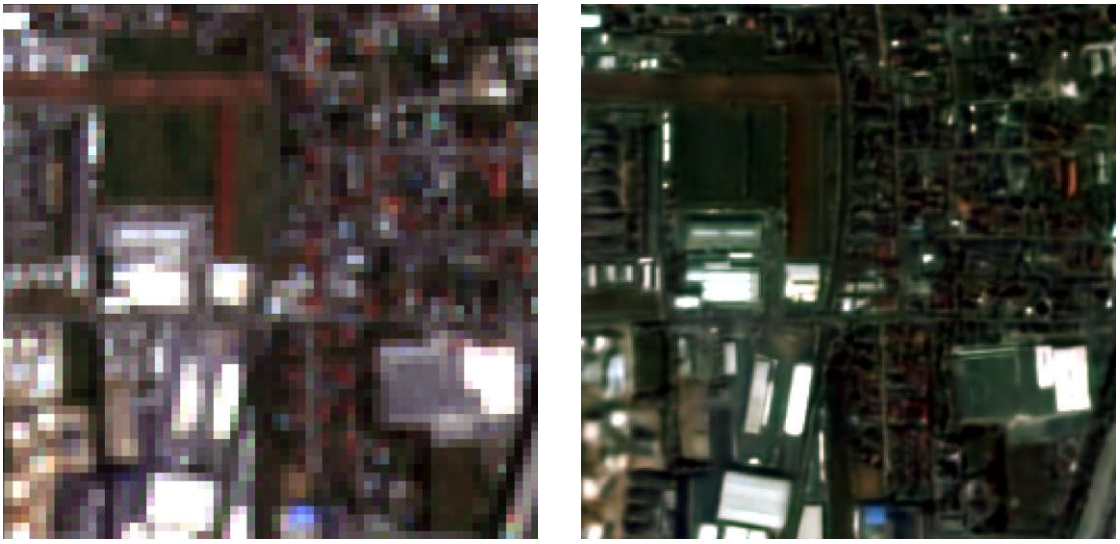
Table 4.10: Prediction averages and found fractions across models, separated into different object sizes. The best results for each category for OFA and OPA are highlighted in bold. Models are sorted in the same order as in Table 4.9.

Name	**0–9**		**10–19**		**20–34**		**35–49**		**50–74**		**75+**	
	Pred	Found	Pred	Found	Pred	Found	Pred	Found	Pred	Found	Pred	Found
Orthophoto Baseline	**0.160**	**0.181**	**0.440**	**0.525**	**0.514**	**0.641**	**0.464**	**0.586**	**0.439**	**0.547**	**0.465**	**0.569**
Bilinear	**0.049**	**0.050**	0.183	**0.190**	0.327	**0.363**	**0.342**	**0.395**	**0.348**	**0.404**	**0.409**	**0.490**
Nearest Neighbour	0.048	0.049	**0.185**	0.189	**0.328**	0.360	0.339	0.385	0.346	0.403	0.400	0.488
Bicubic	0.048	0.048	0.184	**0.190**	0.321	0.354	0.329	0.371	0.334	0.387	0.382	0.462
SEN2SR-RGBN	0.043	0.045	**0.176**	**0.176**	**0.310**	0.327	**0.317**	0.344	**0.323**	**0.367**	**0.376**	**0.457**
SWIN2Mose	0.040	0.040	0.163	0.167	0.306	**0.334**	**0.317**	**0.353**	0.322	0.361	0.373	0.449
Evoland	0.037	0.038	0.166	0.167	0.302	0.327	0.312	0.346	0.314	0.357	0.366	0.442
SEN2SR-Lite	0.041	0.041	0.162	0.163	0.295	0.312	0.304	0.324	0.315	0.352	0.370	0.444
LDSR-S2	**0.047**	**0.047**	0.165	0.165	0.290	0.304	0.302	0.314	0.304	0.323	0.360	0.430
SR4RS	0.040	0.041	0.154	0.155	0.282	0.297	0.302	0.325	0.308	0.340	0.366	0.432
DeepSent	0.039	0.039	0.147	0.149	0.283	0.301	0.296	0.313	0.303	0.331	0.357	0.418

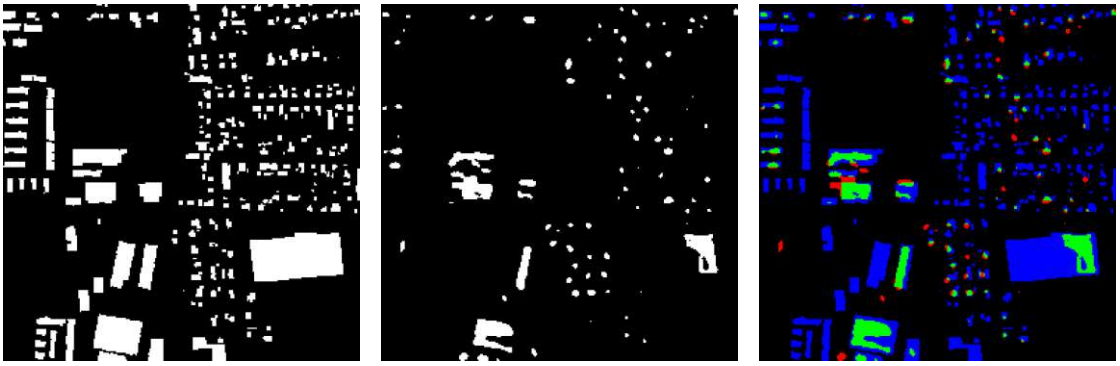
4.3 Application to Proprietary Models

As outlined in Section 3.3.3, the SR model provided by Tracasa could only be applied to a subset of the dataset due to processing limitations. A sample of a Sentinel-2 image and its corresponding Tracasa super-resolved image are presented in Figure 4.11a and 4.11b. A clearly visible spectral shift is introduced in the super-resolved images, likely during data processing steps by Tracasa. Additionally, a spatial shift towards the upper edge of the image can be observed upon close inspection. Apart from these misalignments, urban features are well reconstructed and convincing HR information is introduced. Compared to the SR inference results presented in Section 4.1, Tracasas super-resolved output features less blurring, sharper building borders, and fewer apparent hallucinations. A sample of delineated buildings achieved by applying transfer-learning from a model trained on bilinear interpolated images are presented in Figures 4.11c to 4.11e. Almost no building is correctly identified as a whole, and the majority of buildings are missed completely, indicating an unsuccessful application of transfer-learning.

This poor performance is reflected in the metric results presented in Table 4.11. The results show substantial variation across all evaluated models. Evoland, SEN2SR-Lite, the bilinear baseline, and SEN2SR-RGBN achieved the highest scores, performing at a comparable level. SWIN2Mose and LDSR-S2 followed with noticeably lower results, while SR4RS and DeepSent demonstrated only limited delineation capability, with metric scores around $F_1 = 0.145$. The Tracasa model performed significantly worse than all other approaches. This wide spread of metric values stands in stark contrast to the relatively close performance of models observed in previous evaluations, suggesting inconsistencies in performance and potential limitations in the methodological framework for assessing proprietary SR models.



(a) Sentinel-2 image. (b) Tracasa super-resolved Sentinel-2 image.



(c) GT mask. (d) Prediction map. (e) PCM of the prediction.

Figure 4.11: Sentinel-2 and super-resolved Tracasa image with transfer-learning model results. PCM color-coding: TN black; TP green; FP red; FN blue

Table 4.11: Comparison of transfer-learning applied to super-resolved images on the building segmentation task. The best results for each category for IoU and F_1 are highlighted in bold. Models are sorted by performance in F_1.

Name	Precision	Recall	IoU	F_1
Bilinear	0.516	0.485	**0.330**	**0.485**
Evoland	0.511	0.506	**0.338**	**0.496**
SEN2SR-Lite	0.483	0.511	0.331	0.488
SEN2SR-RGBN	0.491	0.476	0.318	0.472
SWIN2Mose	0.543	0.300	0.231	0.360
LDSR-S2	0.371	0.299	0.196	0.321
SR4RS	0.243	0.126	0.084	0.151
DeepSent	0.245	0.132	0.080	0.139
Tracasa	0.340	0.032	0.030	0.053

CHAPTER 5

Discussion

This chapter reflects on the experimental findings and their broader implications, with the RQs presented in Chapter 1 providing the framework to guide mayor discussion points. Insights are primarily gained from the metric results presented in Section 4.2 and 4.3, and supported by the qualitative assessment of image samples in Section 4.1. Previous work has shown that downstream tasks are essential for evaluating SR. Some studies compare SR outputs directly with LR Sentinel-2 images or other SR methods [8, 14, 11, 2, 17], while others benchmark against native HR data [96, 97]. These literature findings generally agree that: first, downstream evaluation is required to evaluate the performance of SR; second, SR can provide improvements over LR; and third, native HR still performs best. Following this reasoning, comparing different SR models in this thesis helps identify which approaches are most effective for the downstream task of building delineation. Benchmarking against orthophotos highlights the performance gap to true HR images, while interpolated Sentinel-2 offers an additional baseline to isolate the specific benefits of SR over simple resampling by different interpolation methods.

5.1 Discussion of Research Questions

First, it has to be established whether the results obtained from the different sources, HR orthophoto and LR Sentinel-2, are within expected bounds and align with the results of related research. Analysing the absolute performance of building delineation models trained on these datasets shows that the orthophoto-based model achieved the strongest results ($IoU = 0.552$, $F_1 = 0.681$), while Sentinel-2-based ones reached lower values (average $IoU = 0.340$, $F_1 = 0.468$). In domains such as medical imaging, such scores would indicate poor performance [130]. However, given the complexity of the building delineation task from imagery at the comparatively low resolution of 2.5 m, and supported by visual inspection of the outputs, these scores are considered acceptable. Results with similar proportions have been retrieved by Debella-Gilo [87], Panangian and Bittner

[96], and Illarionova et al. [97] on their respective datasets. Furthermore, achieving state-of-the-art absolute performance was not the main objective of this work. Higher scores could likely be obtained through extended experiments with architecture design, fine-tuning, or task-specific optimisations.

Second, it is necessary to assess whether the differences between Sentinel-2–based methods, interpolation, and SR are substantial enough to support meaningful conclusions. If their performance were similar, this would suggest either that the methods are equally effective or that the underlying methodology fails to capture relevant differences. To determine this, the performance gap between the best- and worst-performing models trained on Sentinel-2 imagery can be analysed. It amounts to 0.034 (3.4%) in IoU and 0.040 (4.0%) in F_1. These differences are relatively small; however, other studies comparing building delineation networks report similar ranges of variation [15, 77, 76]. This indicates that an in-depth comparison between interpolation- and SR-based methods is not only viable, but highly beneficial for research purposes, allowing us to analyse and understand their differences. The proposed RQs provide a framework for evaluating and comparing the results achieved in this thesis.

5.1.1 Research Question 1: Are Super-Resolution algorithms advanced enough for their super-resolved output to be used in real-world applications?

The discussion of this RQ covers the general assessment of SR capabilities compared to the orthophoto baseline and interpolation methods. More in-depth analysis is provided in the next RQs. Two different datasets were employed in the experiments: orthophotos and Sentinel-2 images. Orthophotos, derived from VHR sources, inherently contain true HR information, while Sentinel-2 images lack such detail and depend on the insertion of HR details by SR. The performance gap between the two data sources is considerable, amounting to 0.198 (19.8%) in IoU and 0.236 (23.6%) in F_1 for the best performing SR model. Several factors likely contribute to this gap:

1. Native resolution images contain true HR information and are not prone to introduce hallucinations.
2. Orthophotos and GT masks originate from the same source, ensuring perfect pixel alignment, whereas Sentinel-2 data may suffer from spatial and temporal shifts that reduce delineation accuracy.
3. Model fine-tuning was performed on the orthophoto dataset, potentially introducing a bias in its favour.

When comparing interpolation methods to SR, the results indicate that interpolation methods such as bilinear and NN consistently achieve better metric scores than most SR approaches. In only a few cases do SR methods perform equally well, and just one model

(SEN2SR-RGBN) slightly outperforms bicubic interpolation by the small margin of 0.001 in both IoU and F_1. A class-based analysis shows SR methods outperform interpolations only in class 0, which represents areas without buildings. More detail on the class-based analysis is provided in RQ 3 in Section 5.1.3.

This outcome suggests that the additional information introduced by SR is either irrelevant for the downstream task or offset by artefacts such as hallucinations. By contrast, interpolation theoretically increases the spatial resolution without altering the spectral properties of the image nor introducing HR information. Depending on the method, it either smooths the data (bilinear, bicubic) or preserves features more directly (NN). SR models potentially alter the spectral characteristics of the data by super-resolving it, which can introduce errors or change the spectral signature of the image.

Nevertheless, the underlying aim of SR is to enrich images with HR-like detail. The value of this information depends on the implementation and intended purpose. As Wolters, Bastani, and Kembhavi [2] argue, SR should be designed either for human perception (visual quality) or for machine tasks (downstream performance). While some super-resolved images may look visually convincing, they currently fail to outperform interpolation methods in the presented building delineation task. Other SR models not used during the experiments or further advances in SR design may close this gap, but within the scope of this thesis, interpolation remains the stronger baseline.

5.1.2 Research Question 2: Do certain SR algorithms perform better than others, and what impacts these differences?

As shown in detail in Section 4.2, interpolation methods performed almost always better than SR ones. Nevertheless, the differences between evaluated SR models suggests, that some approaches enable more accurate building delineation than others. Various factors likely contribute to the differences in performance between SR models with the three main aspects described below:

- **SR training dataset:** Although all models used Sentinel-2 as the LR input, a main requirement for being eligible for the proposed experiments, the HR reference imagery varied across different sources. These included NAIP (LDSR-S2 and both SEN2SR versions), Spot-6/7 (SR4RS), VENμS (Evoland and SWIN2Mose), Landsat (SWIN2Mose), or synthetic datasets (DeepSent). Differences in the resulting models suggest that the type of reference data may play a role. In particular, the weaker performance of DeepSent, combined with visible speckled noise in its outputs, might indicate limitations when relying solely on synthetic imagery generated through simple image manipulations. By contrast, datasets created through degradation models, as it is partially done for SEN2NAIP, appear to avoid some of these drawbacks.

  A further point of consideration is the spatial coverage of the training data. For example, SEN2NAIP is limited to the continental United States, raising the possi-

bility that models trained on such geographically restricted data may show reduced generalizability when applied to other regions, such as Austria. At the same time, the strong performance of SEN2SR models in this study suggests that spatial restrictions do not necessarily impose a critical drawback, at least in this case. Overall, while these observations hint at potential influences of synthetic training data and geographically constrained datasets, the experiments conducted here do not allow for definitive conclusions.

- **SR output resolution:** The output resolution of the SR models represents another potential influence on performance. Since not all models super-resolved to the same spatial resolution, with Evoland and SWIN2Mose to 5 m, DeepSent to 3.3 m, and the others directly to 2.5 m, a bilinear interpolation was applied to bring all results to a common resolution of 2.5 m. The experiments suggest that this resampling step does not inherently reduce performance, as two of the top-performing models (Evoland and SWIN2Mose) required such interpolation, but still achieved results comparable to models trained at the native 2.5 m scale. This indicates that post-processing interpolation, at least within the tested settings, does not necessarily reduce model performance in downstream tasks and can, in fact support comparability across differing output resolutions.

- **SR model architecture:** The architecture of the SR models could influence downstream performance as well. The best-performing approaches span different model families: the Structured State Space Model (SSM) SEN2SR-RGBN, GAN-based Evoland, transformer-based SWIN2Mose, and the CNN SEN2SR-Lite. Worse performing models, such as SR4RS and LDSR-S2 use generative model backbones, which likely introduced hallucinations, reducing the performance of super-resolved images in the building delineation task. Evoland, although also based on a generative model, outputs at 5 m and is subsequently resampled, which may explain why such artefacts are less pronounced in its results. In contrast, CNN-based models such as DeepSent and SEN2SR-Lite varied in performance, with DeepSent producing particularly noisy outputs, which is likely linked to its reliance on synthetic training data rather than its architecture alone. These observations suggest that while no single model family consistently outperforms the others, the interaction between architecture, training data, and resolution strongly shapes the usefulness of SR outputs for downstream applications.

5.1.3 Research Question 3: Do Super-Resolution algorithms perform better in urban, semi-urban, or rural areas?

Separating evaluation metrics by class provides additional insights into model behaviour beyond overall scores. The analysis of class 0, which contains only non-building areas, provides interesting insights: here, SR models consistently outperform interpolation methods, and both outperform the native HR orthophotos. This suggests that SR approaches can enhance Sentinel-2 imagery without introducing artefacts that are later

misclassified as buildings. While this may seem minor, avoiding FP in areas without built-up structures is highly relevant, as a large proportion of Earth's surface is not urbanised. Robust performance in such cases is especially important for applications such as humanitarian mapping or disaster response, where overestimating buildings can misguide relief efforts.

For the other building-containing classes, different patterns emerge. In classes 1 and 2, sparsely built-up rural areas, which dominate the dataset by contributing the largest proportion of images, small variations between models strongly affect overall performance, making them a critical indicator of general quality. While SR models are capable of correctly predicting many diverse features in these classes, the difference to the interpolation methods is the most significant here among all classes. Small individual buildings, which make up a large proportion of rural areas, are a significant challenge for building delineation models trained on super-resolved images, which is reflected in the metric results and validated by the object-based metrics OPA and OFA.

In classes 3, 4, and 5, which feature more buildings and display semi-urban and urban neighbourhoods, all model performances gradually approach that of models trained on orthophotos. This is likely because a higher density of building pixels reduces the impact of under- or overestimation on segmentation. In these classes, the best SR models outperform some interpolation methods, highlighting the similar performance in such environments. Taken together, these results highlight the value of class-based evaluation: it not only allows a more detailed analysis of where models succeed or struggle but also provides a clearer picture of their robustness across different urban densities, with little additional processing required if such a stratification is already part of the dataset design.

5.1.4 Research Question 4: Can Super-Resolution algorithms adapt to the unique spectral and spatial image composition of Austrian imagery, and can they be applied globally?

To assess whether the selected SR models can generalize from their training datasets to the distinct landscapes of Austria, the results in Table 4.2 provide valuable insight. If SR models fail to generalise, super-resolving Sentinel-2 images would likely introduce hallucinations or artefacts, reducing building segmentation performance compared to interpolation methods. However, the relatively small differences between both approaches suggest that all tested SR models were, to varying degrees, able to handle previously unseen data from Austrian regions.

It should be noted that, to the best of current knowledge, none of the evaluated models were trained specifically on Austrian data. Nevertheless, similarities between Austrian rural and urban settlements and those found elsewhere in western or central Europe may have allowed models to transfer super-resolving capabilities to Austrian imagery. Most training datasets included at least some European coverage, with the notable exception of SEN2NAIP, which is restricted to the continental United States. Despite this spatial

limitation, its broad geographic diversity likely contributed to sufficient generalizability, as indicated by the performance of models trained on it.

To determine whether SR models are globally applicable, more experiments with datasets covering diverse landscapes across all continents have to be conducted. The scope of this thesis, being spatially limited to the extent of Austrian cadastral data and orthophotos, does not allow for concluding about the global generalising capabilities of SR models, but only for landscapes featured in Austria. Addressing this gap requires extending the evaluation of SR models to more varied and geographically distributed case studies, a task that remains open for future research, which can be achieved with the methodology developed within this thesis.

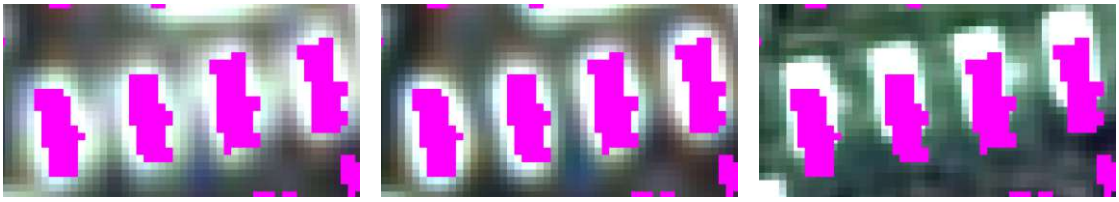
5.2 Further Discussions

The analysis also surfaced additional considerations beyond the formulated RQs, including the evaluation of proprietary SR models and the effectiveness of performance metrics for result interpretation.

5.2.1 Evaluation of Proprietary Models

The application of transfer learning to evaluate SR models on a small batch of images did not yield convincing results. As shown in Table 4.11, performance varied substantially depending on the target super-resolved data it was applied to. These differences suggest that the effectiveness of transfer learning largely depends on the similarity of the target model's outputs to those generated from the bilinear interpolation of Sentinel-2 images, particularly with regard to spatial alignment and spectral characteristics. Notably, Evoland and SEN2SR-Lite even outperformed the bilinear baseline when evaluated on its own set of weights.

As this experiment was conducted solely to evaluate the Tracasa SR model, a particular focus is put on analysing its results here. Two major issues are deemed responsible for its poor performance during the weight transfer. First, the super-resolved Tracasa images exhibited spatial misalignments relative to the GT masks (see Figure 5.1). As the super-resolving of the Tracasa images was completed by the authors of the SR model, it was not possible to correctly identify where and to what extent the spatial shift was introduced. While validating the image pre-processing described in Section 3.3.3.1, no errors in the developed scripts were found. This suggests that the misalignment between the super-resolved images and GT data was introduced during the SR or later processing steps. Second, the spectral distribution of Tracasa outputs deviated considerably from Sentinel-2 and from other SR models that performed better (see Figure 4.11). These combined spatial and spectral discrepancies likely explain the weak results observed in the transfer-learning setup, and overshadow any analysis of the super-resolving capabilities of the Tracasa model.



(a) Sentinel-2 image with GT mask. (b) Evoland image with GT mask. (c) Tracasa image with GT mask.

Figure 5.1: Spatial shift present in the super-resolved Tracasa image with Sentinel-2 and Evoland samples as comparison. GT masks are overlaid in pink.

Overall, these findings highlight that weight transfer from pre-trained models does not provide a reliable or fair framework for comparing SR approaches. The results of this experiment are therefore not used to assess model quality in this thesis. However, the concept remains theoretically feasible: with careful pre-processing and potential spectral harmonisation procedures, transfer learning may still offer a suitable option for evaluating proprietary SR models where full access to inference is restricted. Given the number of private companies developing SR models, such an evaluation framework might allow important insights into the assessment of proprietary models.

5.2.2 Evaluation of the Usability of Used Metric

Not all evaluation metrics contribute equally to assessing building delineation performance. In this thesis, image-based metrics such as IoU and F_1 served as the primary measures, as they capture overall segmentation quality. Their close relationship also allows a redundancy check, ensuring that the implementation of each metric is consistent.

Object-based metrics, including OPA and OFA, were less central but still informative. They correlated well with the image-based metrics, suggesting their general validity. When separated by building size, these metrics highlighted a recurring issue: smaller buildings are detected with much lower reliability than larger ones. However, differences between interpolated and super-resolved images in these smaller classes remained very limited, and the current formulation of the metrics does not fully account for FP predictions. This indicates that further refinement of object-based metrics may be needed to capture performance nuances more effectively, especially for small-scale structures. If an improvement in detecting small objects in super-resolved rather than interpolated images can be verified, it would justify the application of SR, as only SR is capable of injecting high-frequency information in LR images. Such a validation can only be achieved with object-based metrics, which emphasises the need for further development.

CHAPTER 6

Conclusion and Future Works

The research presented throughout this thesis has deepened the understanding of SR in remote sensing, revealing both its potential and its current limitations. The findings highlight how model performance depends not only on visual quality to human observers but on its relevance for downstream applications such as building delineation, underscoring the need for further reliable evaluation strategies.

6.1 Conclusions

This thesis evaluates the potential of SR for improving building delineation from Sentinel-2 imagery and compares it with results obtained from building delineation applied to native HR aerial images. Similar to the research reviewed in Section 2.3.2.2, SR provides an inferior foundation for training building delineation networks compared to native HR images. Furthermore, the results consistently showed that on the tested dataset, SR is outperformed by simpler and less resource-intensive interpolation methods. The NN and bilinear interpolation methods achieved better results than the deployed SR models, while avoiding the risk of introducing artefacts such as hallucinations. Only a handful of SR models, most notably SEN2SR-RGBN and Evoland, came close to matching interpolation-based methods. At present, the added complexity of super-resolving Sentinel-2 imagery to a higher resolution, aiming to improve building delineation results, is not justified. Interpolation methods allow a slightly better result, while not being reliant on GPUs to infer LR Sentinel-2 images.

An important consideration for SR model capabilities is their training dataset's diversity to ensure generalisation. The tested models were able to generalise reasonably well to Austrian landscapes, despite not being explicitly trained on them, but global generalisation could not be confirmed due to the spatially restricted design of this thesis. On the other hand, comparing models by interpolating outputs to a common resolution proved to be a feasible and effective evaluation strategy, allowing for meaningful model comparisons

without introducing significant biases. Furthermore, a method for assessing proprietary models by applying pre-trained weights from interpolation-trained models was developed. However, this method did not prove successful, as biases towards the interpolated image data outweighed any possible discernible difference between model performances.

The experiments revealed that not all SR models behave alike. While some approaches obtained poorer results, others achieved results close to interpolation methods. Because many factors shape how super-resolved images perform in the building delineation task, it is difficult to pinpoint which ones dominate the outcome. The results suggest, however, that three key design choices have the largest influence. First, the dataset used for training SR models largely defines its ability to generalise and adapt to new inputs. Models trained on cross-sensor data proved more reliable than those relying on synthetic datasets created through basic image manipulations. Second, the choice of model architecture affects susceptibility to hallucinations and spectral shifts. In particular, generative architectures appear to be more prone to introducing artefacts which harm the performance on building delineation metrics, even though they often produce outputs that are visually more appealing to human observers, an aspect not further analysed in this thesis. Third, the target output resolution also matters. Models producing images at a lower resolution (e.g., 5 m instead of 2.5 m) tend to introduce fewer artefacts, as the SR factor is smaller. This reduces the negative effects of hallucinations on downstream tasks, provided that subsequent interpolation ensures comparability across resolutions. These insights can guide future SR development to create reliable models and provide a reference framework to assess their downstream task performance on building delineation.

6.2 Thesis Outcomes Beyond Enhanced Scientific Knowledge

This thesis's core objective, the evaluation of SR algorithms on the downstream task of building delineation, was achieved, and complemented by the completion of additional contributions. These include the creation of the *austriadownloader* package, enabling streamlined access to Austrian orthophotos and cadastral masks. Released as a Python package, it is openly available for use and further development by the research community. In addition, all datasets used in this work, including orthophotos, cadastral masks, Sentinel-2 imagery, and the generated super-resolved Sentinel-2 datasets, are published in an online repository, together with the full code base under an open-source license: https://github.com/Zerhigh/Evaluating_Sentinel-2_Super-Resolution_Algorithms_for_Automated_Building_Delineation. This ensures transparency, reproducibility, and the possibility of extending the approach with alternative building delineation models and processing techniques.

These results were made possible through close collaboration and knowledge exchange with colleagues at the University of Valencia during a research stay. Joint work extended to the development of the *austriadownloader* package, contributions to the ESA OpenSR project (https://opensr.eu), and the preparation of datasets in TACO format

(https://tacofoundation.github.io) for DL applications. The stay not only advanced the progress of this thesis but also established the foundation for long-term collaboration between the University of Valencia and the Technical University of Vienna.

Parts of this work, particularly the *austriadownloader* package, have already been presented at the AGIT conference in Salzburg and are published in its proceedings [99]. The main outcomes of the thesis will also be prepared for submission to a peer-reviewed scientific journal.

6.3 Future Work

The results of this thesis answer the core RQs, which show that interpolation methods provide better performance on the downstream task of building delineation than current state-of-the-art SR models. This highlights many open directions for further investigation. The broader applicability of SR remains only partly explored, and additional research is needed to improve upon the robust foundation for grading, assessing, and comparing both current and future SR algorithms.

A key step will be the formalisation of evaluation procedures within the SR research community. Current SR implementations vary widely in their design choices, including LR inputs, HR references, target resolutions, and whether single- or multi-image approaches are used. Without a shared taxonomy, comparisons between models remain inconsistent. Developing and agreeing on such a taxonomy would provide a standardised framework for classifying SR approaches and enable fairer evaluations across the field.

In addition, a diverse set of downstream tasks should be established to assess SR performance across the full spectrum of Earth observation applications. These tasks should include multiple sensors (from optical to microwave), and a variety of methodologies, ranging from non-DL techniques (e.g., spectral index classification) to DL-based feature extraction (e.g., building and field delineation) and modelling approaches (e.g., leaf-area index estimation). Creating such a benchmark requires close collaboration with domain experts to identify meaningful applications, provide high-quality data sources, and ensure reliability of both LR and HR datasets. Importantly, the benchmark must include globally distributed downstream tasks to assess generalisation, similar to the approach of the PANGAEA benchmark for geospatial foundation models [88].

This thesis contributes only a small part to this larger vision: providing a framework for evaluating SR through the task of building delineation. Extending such frameworks will not only strengthen scientific validation but also democratize access to high-quality Earth observation data, enabling institutions with limited resources to benefit from super-resolved imagery.

CHAPTER 7

Appendices

Listing 7.1: HRNet model configuration

```
DATASET:
  NUM_CLASSES: 1
MODEL:
  NAME: seg_hrnet_ocr
  NUM_OUTPUTS: 2
  PRETRAINED: None
  ALIGN_CORNERS: True
  OCR:
    MID_CHANNELS: 512
    KEY_CHANNELS: 256
    DROPOUT: 0.05
    SCALE: 1
  EXTRA:
    FINAL_CONV_KERNEL: 1
    STAGE1:
      NUM_MODULES: 1
      NUM_BRANCHES: 1
      BLOCK: BOTTLENECK
      NUM_BLOCKS:
      - 4
      NUM_CHANNELS:
      - 64
      FUSE_METHOD: SUM
    STAGE2:
      NUM_MODULES: 1
      NUM_BRANCHES: 2
      BLOCK: BASIC
      NUM_BLOCKS:
      - 4
```

```
      - 4
      NUM_CHANNELS:
      - 48
      - 96
      FUSE_METHOD: SUM
    STAGE3:
      NUM_MODULES: 4
      NUM_BRANCHES: 3
      BLOCK: BASIC
      NUM_BLOCKS:
      - 4
      - 4
      - 4
      NUM_CHANNELS:
      - 48
      - 96
      - 192
      FUSE_METHOD: SUM
    STAGE4:
      NUM_MODULES: 3
      NUM_BRANCHES: 4
      BLOCK: BASIC
      NUM_BLOCKS:
      - 4
      - 4
      - 4
      - 4
      NUM_CHANNELS:
      - 48
      - 96
      - 192
      - 384
      FUSE_METHOD: SUM
```

List of Figures

List of Tables

Acronyms

API Application Programming Interface. 22, 39, 41, 46, 55

AVIRIS Airborne Visible/Infrared Imaging Spectrometer. 11, 13

AWS Amazon Web Services. 22

BCE Binary Cross-Entropy. 43, 44, 49, 52

BEV Bundesamt für Eich- und Vermessungswesen. 3, 23, 24, 35

BOA Bottom-of-Atmosphere. 21

CDF Cumulative Probability Distribution Function. 31

CNN Convolutional Neural Network. 7, 11–13, 15, 36, 42, 88

COG Cloud Optimized GeoTIFF. 22, 23, 28, 30

CRS Coordinate Reference System. 22, 24, 26–28, 30

CSS Cloud Cover Score. 29, 30

DEM Digital Elevation Model. 23

DKM Digital Cadastral Map (German: Digitale Katastralmappe). 24

DL Deep Learning. 2, 5–8, 10, 12, 14–17, 22, 24, 26, 28, 31, 39, 41, 42, 45, 95

ESPCN Efficient Sub-Pixel Convolutional Neural Network. 18

ESRGAN Enhanced Super-Resolution Generative Adversarial Networks. 7, 10, 13, 18

FN False Negative. 43, 61, 63–72, 83, 99, 100

FP False Positive. 43, 48, 61–72, 83, 89, 91, 99, 100

FTL Focal Tversky Loss. 43, 44, 49, 51–53, 55, 101

GAN Generative Adversarial Networks. 7, 10, 13, 88

GEE Google Earth Engine. 3, 22, 29, 30

GPU Graphics Processing Unit. 39, 45, 93

GSD Ground Sampling Distance. 22, 28

GT Ground Truth. 14, 15, 17, 18, 24, 26, 28, 29, 35, 44, 47, 48, 61, 63–72, 83, 86, 90, 91, 100

HR High Resolution. 2, 3, 6–11, 13–19, 26, 29, 35, 39, 61, 82, 85–88, 93, 95

HRNet High-Resolution Net. 15–17, 42, 43, 49–52, 101

IoU Intersection over Union. 15, 18, 42, 46, 47, 50–53, 73–79, 84–87, 91, 101, 102

L1B Level-1B. 36

L1C Level-1C. 21, 22, 36

L2A Level-2A. 9, 21, 22, 29, 35

LLM Large Language Model. 26

LPIPS Learned Perceptual Image Patch Similarity. 7, 13, 18

LR Low Resolution. 3, 6–12, 18, 26, 29, 45, 85, 87, 91, 93, 95

LUT Lookup Table. 26, 28

ML Machine Learning. 6

MSE Mean Squared Error. 7, 13

MSI MultiSpectral Instrument. 11, 21

NAIP National Agriculture Imagery Program. 9, 13, 18, 87

NIR Near-Infrared. 1, 3, 9, 10, 12, 13, 23, 24, 28, 36

NN Nearest Neighbor. 5, 30, 45, 58, 60, 62, 63, 73, 74, 80, 86, 87, 93, 99

OCR Object-Contextual Representations. 16, 42, 43, 49, 51

OFA Object Found Average. 48, 73, 80, 81, 89, 91, 102

OPA Object Prediction Average. 48, 73, 80, 81, 89, 91, 102

OSM Open Street Map. 18

PCM Pixel-wise Confusion Map. 61–72, 83, 99, 100

PSNR Peak Signal-to-Noise Ratio. 7, 10, 11, 13, 18

RAM Random-Access Memory. 45

RGB Red, Green, and Blue. 1, 3, 9, 10, 12, 13, 21, 23, 28, 36, 57

RQ Research Question. 1–3, 57, 85–87, 90, 95

SAM Segment Anything Model. 15, 18, 42

SAN Second-order Attention Network. 13

SE Squeeze-and-Excitation. 16, 42

smp segmentation-models-pytorch. 26, 42

SR Super-Resolution. 1–3, 5–14, 17–19, 21, 22, 26, 28, 31, 35–37, 39–42, 44, 45, 53, 55, 57, 60–62, 66–74, 80, 82, 85–91, 93–95, 99, 101

SRCNN Super-Resolution Convolutional Neural Network. 7

SSIM Structural Similarity Index Measure. 11, 13, 18

SSM Structured State Space Model. 88

SWIN Shifted Window Transformer. 11–13, 18

SWIR Short Wave Infra-Red. 12, 23

timm Pytorch Image Models. 26, 42

TN True Negative. 61, 63–72, 83, 99, 100

TOA Top-of-Atmosphere. 21

TP True Positive. 47, 61, 63–72, 83, 99, 100

UAV Unmanned Aerial Vehicle. 14, 15, 22

UTM Universal Transverse Mercator. 22

VHR Very High Resolution. 1, 3, 14–19, 22, 86

wandb Weights & Biases. 26, 46, 55

Bibliography

[1] BEV. *Serie Digitales Orthophoto Farbe und Infrarot (DOP RGBI) Stichtag 25.06.2024*. Accessed: 20 Feb 2025. 2024. DOI: 10.48677/f2e11a84-cdc7-4cfa-b048-da3675d58704.

[2] Piper Wolters, Favyen Bastani, and Aniruddha Kembhavi. „Zooming out on zooming in: Advancing super-resolution for remote sensing". In: *arXiv preprint arXiv:2311.18082* (2023).

[3] Pawel Kowaleczko et al. „A real-world benchmark for Sentinel-2 multi-image super-resolution". In: *Scientific Data* 10.1 (2023), p. 644.

[4] Tomasz Tarasiewicz et al. „Multitemporal and multispectral data fusion for super-resolution of Sentinel-2 images". In: *IEEE Transactions on Geoscience and Remote Sensing* 61 (2023), pp. 1–19.

[5] Muhammad Sarmad, Arnt-Børre Salberg, and Michael Kampffmeyer. „DiffFuSR: Super-Resolution of all Sentinel-2 Multispectral Bands using Diffusion Models". In: *arXiv preprint arXiv:2506.11764* (2025).

[6] Charis Lanaras et al. „Super-resolution of Sentinel-2 images: Learning a globally applicable deep neural network". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 146 (2018), pp. 305–319.

[7] Yi Xiao et al. „EDiffSR: An efficient diffusion probabilistic model for remote sensing image super-resolution". In: *IEEE Transactions on Geoscience and Remote Sensing* 62 (2023), pp. 1–14.

[8] Aurélien Lac et al. „Sentinel-2 Single Image Super-Resolution with the SEN2VENμS Dataset: architecture, training strategy, performances assessment and application to Water Bodies Detection". Submitted to IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. Sept. 2023. URL: https://hal.science/hal-04218629.

[9] Ngoc Long Nguyen et al. „L1BSR: Exploiting detector overlap for self-supervised single-image super-resolution of sentinel-2 L1B imagery". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 2013–2023.

[10] Simon Donike et al. „Trustworthy Super-Resolution of Multispectral Sentinel-2 Imagery with Latent Diffusion“. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* (2025).

[11] Cesar Aybar et al. „A Radiometrically and Spatially Consistent Super-Resolution Framework for Sentinel-2“. In: *Available at SSRN 5247739* (2025).

[12] Aditya Retnanto et al. „Beyond Pretty Pictures: Combined Single-and Multi-Image Super-resolution for Sentinel-2 Images“. In: *arXiv preprint arXiv:2505.24799* (2025).

[13] Rémi Cresson. „SR4RS: A tool for super resolution of remote sensing images“. In: *Journal of Open Research Software* 10.1 (2022).

[14] Leonardo Rossi et al. „Swin2-MoSE: A new single image supersolution model for remote sensing“. In: *IET Image Processing* 19.1 (2025), e13303.

[15] Christian Ayala et al. „A deep learning approach to an enhanced building footprint and road detection in high-resolution satellite imagery“. In: *Remote Sensing* 13.16 (2021), p. 3135.

[16] Muhammed T Razzak et al. „Multi-spectral multi-image super-resolution of Sentinel-2 with radiometric consistency losses and its effect on building delineation“. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 195 (2023), pp. 1–13.

[17] Nejc Vesel et al. *Multi-temporal Super-Resolution on Sentinel-2 Imagery.* “Planet Stories” – EO Research blog, part of the DIONE project by Sinergise. May 2021.

[18] Cesar Aybar et al. „A Comprehensive Benchmark for Optical Remote Sensing Image Super-Resolution“. In: *IEEE Geoscience and Remote Sensing Letters* (2024).

[19] PR Smith. „Bilinear interpolation of digital images“. In: *Ultramicroscopy* 6.2 (1981), pp. 201–204.

[20] Maganti Jahnavi, D Rajeswara Rao, and Amballa Sujatha. „A comparative study of super-resolution interpolation techniques: Insights for selecting the most appropriate method“. In: *Procedia Computer Science* 233 (2024), pp. 504–517.

[21] Donya Khaledyan et al. „Low-cost implementation of bilinear and bicubic image interpolation for real-time image super-resolution“. In: *2020 IEEE Global Humanitarian Technology Conference (GHTC)*. IEEE. 2020, pp. 1–5.

[22] Tao Zhang et al. „FSRSS-Net: High-resolution mapping of buildings from middle-resolution satellite images using a super-resolution semantic segmentation network“. In: *Remote Sensing* 13.12 (2021), p. 2290.

[23] Juan Mario Haut et al. „Remote sensing image superresolution using deep residual channel attention“. In: *IEEE Transactions on Geoscience and Remote Sensing* 57.11 (2019), pp. 9277–9289.

[24] Mikel Galar et al. „Super-resolution of sentinel-2 images using convolutional neural networks and real ground truth data“. In: *Remote Sensing* 12.18 (2020), p. 2941.

[25] Dianyuan Han. „Comparison of commonly used image interpolation methods“. In: *Conference of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)*. Atlantis Press. 2013, pp. 1556–1559.

[26] Marco Venturelli. *The Dangers Behind Image Resizing*. Accessed: 2025-05-23. Aug. 2021. URL: https://zuru.tech/blog/the-dangers-behind-image-resizing.

[27] Fredrik Lundh and Jeffrey A. Clark. *Pillow (PIL Fork) Documentation*. Accessed: 2025-05-23. 2025. URL: https://buildmedia.readthedocs.org/media/pdf/pillow/latest/pillow.pdf.

[28] Sina Farsiu et al. „Advances and challenges in super-resolution“. In: *International Journal of Imaging Systems and Technology* 14.2 (2004), pp. 47–57.

[29] Sina Farsiu et al. „Fast and robust multiframe super resolution“. In: *IEEE transactions on image processing* 13.10 (2004), pp. 1327–1344.

[30] Daniel Glasner, Shai Bagon, and Michal Irani. „Super-resolution from a single image“. In: *2009 IEEE 12th international conference on computer vision*. IEEE. 2009, pp. 349–356.

[31] Chao Dong et al. „Learning a deep convolutional network for image super-resolution“. In: *European conference on computer vision*. Springer. 2014, pp. 184–199.

[32] Mukhriddin Arabboev et al. „A comprehensive review of image super-resolution metrics: classical and AI-based approaches“. In: *Acta IMEKO* 13.1 (2024), pp. 1–8.

[33] Xintao Wang et al. „ESRGAN: Enhanced super-resolution generative adversarial networks“. In: *Proceedings of the European conference on computer vision (ECCV) workshops*. Sept. 2018.

[34] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. „Perceptual losses for real-time style transfer and super-resolution“. In: *European conference on computer vision*. Springer. 2016, pp. 694–711.

[35] Robin Rombach et al. „High-resolution image synthesis with latent diffusion models“. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 10684–10695.

[36] Richard Zhang et al. „The unreasonable effectiveness of deep features as a perceptual metric“. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 586–595.

[37] Michel Deudon et al. „Highres-net: Recursive fusion for multi-frame super-resolution of satellite imagery“. In: *arXiv preprint arXiv:2002.06460* (2020).

[38] Christian Ayala et al. „Tiny Object Detection in Super-Resolved Sentinel-2 Imagery“. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 48 (2025), pp. 61–66.

[39] Peijuan Wang, Bulent Bayram, and Elif Sertel. „A comprehensive review on deep learning based remote sensing image super-resolution methods“. In: *Earth-Science Reviews* 232 (2022), p. 104110.

[40] Tomasz Tarasiewicz et al. „Multitemporal and multispectral data fusion for super-resolution of Sentinel-2 images“. In: *IEEE Transactions on Geoscience and Remote Sensing* 61 (2023), pp. 1–19.

[41] Cesar Aybar et al. „SEN2NAIP: A large-scale dataset for Sentinel-2 Image Super-Resolution“. In: *Scientific Data* 11.1 (2024), p. 1389.

[42] Julien Cornebise, Ivan Oršolić, and Freddie Kalaitzis. „Open high-resolution satellite imagery: The worldstrat dataset–with application to super-resolution“. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 25979–25991.

[43] Marcus Märtens et al. „Super-resolution of PROBA-V images using convolutional neural networks“. In: *Astrodynamics* 3.4 (2019), pp. 387–402.

[44] Julien Michel et al. „Sen2venμs, a dataset for the training of sentinel-2 super-resolution algorithms“. In: *Data* 7.7 (2022), p. 96.

[45] Manuel Grizonnet et al. „Orfeo ToolBox: open source processing of remote sensing images“. In: *Open Geospatial Data, Software and Standards* 2.1 (2017), p. 15.

[46] Christian Ledig et al. „Photo-realistic single image super-resolution using a generative adversarial network“. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4681–4690.

[47] Francesco Salvetti et al. „Multi-image super resolution of remotely sensed images using residual attention deep neural networks“. In: *Remote Sensing* 12.14 (2020), p. 2207.

[48] Junwei Wang et al. „Multisensor remote sensing imagery super-resolution with conditional GAN“. In: *Journal of Remote Sensing* (2021).

[49] Dominik Koßmann, Viktor Brack, and Thorsten Wilhelm. „Seasonet: A seasonal scene classification, segmentation and retrieval dataset for satellite imagery over germany“. In: *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 2022, pp. 243–246.

[50] Cesar Aybar et al. „CloudSEN12, a global dataset for semantic understanding of cloud and cloud shadow in Sentinel-2“. In: *Scientific data* 9.1 (2022), p. 782.

[51] Cesar Aybar et al. „Lessons learned from CloudSEN12 dataset: identifying incorrect annotations in cloud semantic segmentation datasets“. In: *IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 2023, pp. 892–895.

[52] Cesar Aybar et al. „CloudSEN12+: The largest dataset of expert-labeled pixels for cloud and cloud shadow detection in Sentinel-2“. In: *Data in Brief* 56 (2024), p. 110852.

[53] Enrique Portalés-Julià et al. „Global flood extent segmentation in optical satellite images“. In: *Scientific reports* 13.1 (2023), p. 20316.

[54] Anna Vaughan et al. „AI for operational methane emitter monitoring from space“. In: *arXiv preprint arXiv:2408.04745* (2024).

[55] Juan Francisco Amieva, Christian Ayala, and Mikel Galar. „Super-resolution of Sentinel-1 Imagery Using an Enhanced Attention Network and Real Ground Truth Data“. In: *Proceedings of SPAICE2024: The First Joint European Space Agency/IAA Conference on AI in and for Space*. 2024, pp. 198–203.

[56] Juan Francisco Amieva, Christian Ayala, and Mikel Galar. „A deep learning approach to jointly super-resolve and despeckle Sentinel-1 imagery“. In: *Acta Astronautica* (2025).

[57] Christian Ayala, Carlos Aranda, and Mikel Galar. „Towards fine-grained road maps extraction using Sentinel-2 imagery“. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 3 (2021), pp. 9–14.

[58] Tao Dai et al. „Second-order Attention Network for Single Image Super-Resolution“. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 11065–11074.

[59] Alina Kuznetsova et al. „The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale“. In: *International journal of computer vision* 128.7 (2020), pp. 1956–1981.

[60] Anka Reuel-Lamparth et al. „Betterbench: Assessing ai benchmarks, uncovering issues, and establishing best practices“. In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 21763–21813.

[61] Daniel Kostrzewa et al. „B4MultiSR: a benchmark for multiple-image super-resolution reconstruction“. In: *International Conference: Beyond Databases, Architectures and Structures*. Springer. 2018, pp. 361–375.

[62] Ribana Roscher et al. „Better, not just more: Data-centric machine learning for earth observation“. In: *IEEE Geoscience and Remote Sensing Magazine* (2024).

[63] Maximilien Houël et al. „AI-Based Super Resolution in Climate Crisis Context“. In: *IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 2023, pp. 1660–1663.

[64] Nicholas Weir et al. „Spacenet mvoi: A multi-view overhead imagery dataset“. In: *Proceedings of the ieee/cvf international conference on computer vision*. 2019, pp. 992–1001.

[65] Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow. „Spacenet: A remote sensing dataset and challenge series“. In: *arXiv preprint arXiv:1807.01232* (2018).

[66] Ronny Hänsch et al. „The spacenet 8 challenge-from foundation mapping to flood detection“. In: *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 2022, pp. 5073–5076.

[67] Shunping Ji, Shiqing Wei, and Meng Lu. „Fully convolutional networks for multi-source building extraction from an open aerial and satellite imagery data set“. In: *IEEE Transactions on geoscience and remote sensing* 57.1 (2018), pp. 574–586.

[68] Shiqing Wei et al. „BuildMapper: A fully learnable framework for vectorized building contour extraction“. In: *ISPRS journal of photogrammetry and remote sensing* 197 (2023), pp. 87–104.

[69] Volodymyr Mnih. *Machine learning for aerial image labeling*. University of Toronto (Canada), 2013.

[70] Nelson Nauata and Yasutaka Furukawa. „Vectorizing world buildings: Planar graph reconstruction by primitive detection and relationship inference“. In: *European Conference on Computer Vision*. Springer. 2020, pp. 711–726.

[71] OpenStreetMap contributors. *About OpenStreetMap*. `https://wiki.openstreetmap.org/wiki/About_OpenStreetMap`. Accessed: 2025-08-22. 2025.

[72] D Koc San and M Turker. „Automatic building detection and delineation from high resolution space images using model-based approach“. In: *Proceedings of the ISPRS workshop on topographic mapping from space*. 2006.

[73] Chungan Lin and Ramakant Nevatia. „Building detection and description from a single intensity image“. In: *Computer vision and image understanding* 72.2 (1998), pp. 101–121.

[74] R Bruce Irvin and David M McKeown. „Methods for exploiting the relationship between buildings and their shadows in aerial imagery“. In: *IEEE Transactions on Systems, Man, and Cybernetics* 19.6 (1989), pp. 1564–1575.

[75] Shiqing Wei et al. „From lines to polygons: Polygonal building contour extraction from high-resolution remote sensing imagery“. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 209 (2024), pp. 213–232.

[76] Ziming Li et al. „A deep learning-based framework for automated extraction of building footprint polygons from very high-resolution aerial imagery“. In: *Remote Sensing* 13.18 (2021), p. 3630.

[77] Yuxuan Li et al. „HD-Net: High-resolution decoupled network for building footprint extraction via deeply supervised body and boundary decomposition“. In: *ISPRS journal of photogrammetry and remote sensing* 209 (2024), pp. 51–65.

[78] Emmanuel Maggiori et al. „Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark“. In: *2017 IEEE International geoscience and remote sensing symposium (IGARSS)*. IEEE. 2017, pp. 3226–3229.

[79] Jingdong Wang et al. „Deep high-resolution representation learning for visual recognition“. In: *IEEE transactions on pattern analysis and machine intelligence* 43.10 (2020), pp. 3349–3364.

[80] Adam Van Etten et al. „The multi-temporal urban development spacenet dataset“. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 6398–6407.

[81] Lucas Prado Osco et al. „The segment anything model (sam) for remote sensing applications: From zero to one shot“. In: *International Journal of Applied Earth Observation and Geoinformation* 124 (2023), p. 103540.

[82] Alexander Kirillov et al. „Segment anything“. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2023, pp. 4015–4026.

[83] Christian Ayala et al. „Diffusion models for remote sensing imagery semantic segmentation“. In: *IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 2023, pp. 5654–5657.

[84] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. „U-net: Convolutional networks for biomedical image segmentation“. In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[85] Penglei Xu et al. „ESPC_NASUnet: An end-to-end super-resolution semantic segmentation network for mapping buildings from remote sensing images“. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14 (2021), pp. 5421–5435.

[86] Vuong Nguyen et al. „Building footprint extraction in dense areas using super resolution and frame field learning“. In: *2023 12th International Conference on Awareness Science and Technology (iCAST)*. IEEE. 2023, pp. 112–117.

[87] Misganu Debella-Gilo. „Relative performance of super-resolved Sentinel-2 and Copernicus VHR images in mapping built-up areas and building footprints using deep learning“. In: *European Journal of Remote Sensing* 58.1 (2025), p. 2517381.

[88] Valerio Marsocci et al. „Pangaea: A global and inclusive benchmark for geospatial foundation models“. In: *arXiv preprint arXiv:2412.04204* (2024).

[89] Kaiming He et al. „Deep residual learning for image recognition“. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[90] Saining Xie et al. „Aggregated residual transformations for deep neural networks“. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1492–1500.

[91] Jie Hu, Li Shen, and Gang Sun. „Squeeze-and-excitation networks“. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7132–7141.

[92] Lin Feng et al. „National-scale mapping of building footprints using feature super-resolution semantic segmentation of Sentinel-2 images“. In: *GIScience & Remote Sensing* 60.1 (2023), p. 2196154.

[93] Zeping Liu et al. „China Building Rooftop Area: the first multi-annual (2016–2021) and high-resolution (2.5 m) building rooftop area dataset in China derived with super-resolution segmentation from Sentinel-2 imagery“. In: *Earth System Science Data* 15.8 (2023), pp. 3547–3572.

[94] Wojciech Sirko et al. „High-resolution building and road detection from sentinel-2“. In: *arXiv preprint arXiv:2310.11622* (2023).

[95] Zhiling Guo et al. „Enhancing Building Semantic Segmentation Accuracy with Super Resolution and Deep Learning: Investigating the Impact of Spatial Resolution on Various Datasets“. In: *arXiv preprint arXiv:2307.04101* (2023).

[96] Daniel Panangian and Ksenia Bittner. „Can Location Embeddings Enhance Super-Resolution of Satellite Imagery?“ In: *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2025, pp. 6136–6145.

[97] Svetlana Illarionova et al. „Benchmark for building segmentation on up-scaled Sentinel-2 imagery“. In: *Remote Sensing* 15.9 (2023), p. 2347.

[98] Darius Lam et al. „xview: Objects in context in overhead imagery“. In: *arXiv preprint arXiv:1802.07856* (2018).

[99] Samuel Hollendonner et al. „AustriaDownloader: A Processing Pipeline for Merging Austrian Orthophotos and Cadastral Data“. In: *Shaping Geospatial Futures*. Ed. by Johannes Scholz. Z_GIS, Universität Salzburg, 2025, pp. 88–93. DOI: https://doi.org/10.25598/agit/2025-15. URL: https://eplus.uni-salzburg.at/agit/periodical/titleinfo/12103759?.

[100] European Space Agency. *Sentinel-2 Mission - SentiWiki*. https://sentiwiki.copernicus.eu/web/s2-mission. Accessed: 2025-05-22. 2024. URL: https://sentiwiki.copernicus.eu/web/s2-mission.

[101] European Space Agency. *Sentinel-2 Products Specification Document*. Tech. rep. S2-PDGS-CS-DI-PSD. Version 15.0. Accessed: 2025-05-23. European Space Agency, Apr. 2024. URL: https://sentinels.copernicus.eu/documents/d/sentinel/s2-pdgs-cs-di-psd-v15-0.

[102] Bernhard Bauer-Marschallinger and Konstantin Falkner. „Wasting petabytes: A survey of the Sentinel-2 UTM tiling grid and its spatial overhead“. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 202 (2023), pp. 682–690.

[103] Copernicus Data Space Ecosystem. *Quotas and Limitations*. https://documentation.dataspace.copernicus.eu/Quotas.html. Accessed: 2025-05-23. 2024.

[104] Element 84. *Sentinel-2 Level-2A Cloud-Optimized GeoTIFFs*. https://registry.opendata.aws/sentinel-2-l2a-cogs/. Accessed: 2025-05-23. 2024.

[105] European Space Agency (ESA). *Copernicus Sentinel-2 Surface Reflectance*. https://developers.google.com/earth-engine/datasets/catalog/COPERNICUS_S2_SR_HARMONIZED. Accessed: 2025-04-24. 2022.

[106] Lei Ma et al. „Deep learning in remote sensing applications: A meta-analysis and review“. In: *ISPRS J. Photogramm. Remote Sens.* 152 (2019), pp. 166–177.

[107] Klaus Steinnocher, Gebhard Banko, and Jürgen Weichselbaum. *Planungsrelevante Datengrundlagen für Österreich: LISA–Land Information System Austria*. na, 2011.

[108] BEV. *Orthophoto Farbe*. https://www.bev.gv.at/Services/Produkte/Luftbildprodukte/Orthophoto-Farbe.html. Accessed: 17 Feb 2025. 2025.

[109] Open Geospatial Consortium. *OGC Cloud Optimized GeoTIFF Standard*. http://www.opengis.net/doc/is/COG/1.0. Accessed: 2025-02-19. 2023.

[110] Michelle D'Arcy, Marina Nistotskaya, and Robert Ellis. „Mapping the state: Measuring infrastructural power through cadastral records". In: *Proceedings of the Int. Federation of Surveyors' Working Week*. 9784 (2019).

[111] Reinfried Mansberger et al. „The Characteristics of the Austrian Cadastre". In: *Kart og Plan* 117.2 (2024), pp. 182–190.

[112] BEV. *Katastralmappe und Sachdaten digital*. https://www.bev.gv.at/Services/Produkte/Kataster-und-Verzeichnisse/Katastralmappe-und-Sachdaten-digital.html. Accessed: 17 Feb 2025. 2025.

[113] BEV. *Digitale Katastralmappe und Grundstücksdaten Stichtagsdaten GPKG*. https://data.bev.gv.at/download/Kataster/gpkg/national/BEV_S_KA_Katastralmappe_Grundstuecksdaten_GPKG_V1.0.pdf. Accessed: 18 Feb 2025. 2023.

[114] Open Geospatial Consortium. *OGC GeoPackage Encoding Standard 1.3*. http://www.opengis.net/doc/IS/geopackage/1.3. Accessed: 2025-02-19. 2021.

[115] BEV. *Kataster Grafik Grundstücksverzeichnis GPKG Stichtag 01.04.2021*. Accessed: 18 Feb 2025. 2021. DOI: 10.48677/2447c4db-95fc-4163-9df2-d7cedf16e210.

[116] BEV. *Kataster Grafik Grundstücksverzeichnis GPKG Stichtag 01.04.2022*. Accessed: 18 Feb 2025. 2022. DOI: 10.48677/bc5d6609-8f75-43cb-95f9-9512abf12485.

[117] BEV. *Kataster Grafik Grundstücksverzeichnis GPKG Stichtag 01.04.2023*. Accessed: 18 Feb 2025. 2023. DOI: 10.48677/7f2ce3ef-bf81-4cfe-ac01-b01b58161e85.

[118] George F Jenks. „The data model concept in statistical mapping". In: *International yearbook of cartography* 7 (1967), pp. 186–190.

[119] S2 MSI ESL Team. *Data Quality Report – Sentinel-2 Level-1C MSI, January 2025*. Tech. rep. OMPC.CS.DQR.01.12-2024. Version Issue 107.0. Copernicus Space Component Sentinel Optical Imaging Mission Performance Cluster Service, Jan. 2025. URL: https://sentiwiki.copernicus.eu/__attachments/1673423/OMPC.CS.DQR.001.12-2024%20-%20MSI%20L1C%20DQR%20January%202025%20-%20107.0.pdf.

[120] Samuel Hollendonner, Negar Alinaghi, and Ioannis Giannopoulos. „Road Network Mapping from Multispectral Satellite Imagery: Leveraging Deep Learning and Spectral Bands". In: *AGILE: GIScience Series* 5 (2024), p. 6.

[121] Yosef Akhtman. *Sentinel-2 Deep Resolution 3.0: Effective 12-Band 10× Single-Image Super-Resolution for Sentinel-2*. Oct. 2023. URL: https://medium.com/@ya_71389/sentinel-2-deep-resolution-3-0-c71a601a2253.

[122] Eugene Siow. *super-image: Image super resolution models for PyTorch*. https://github.com/eugenesiow/super-image. GitHub repository, Apache-2.0 license, 180 stars. July 2021.

[123] Image and Signal Processing (ISP) at UV-ES. *superIX: SUPERIX Super-Resolution Intercomparison Exercise model*. Hugging Face model repository. Available at https://huggingface.co/isp-uv-es/superIX, licensed under Apache-2.0. 2024.

[124] HRNet Team. *HRNet-Semantic-Segmentation: HRNet + OCR branch*. https://github.com/HRNet/HRNet-Semantic-Segmentation/tree/HRNet-OCR. 2020.

[125] Nabila Abraham and Naimul Mefraz Khan. „A Novel Focal Tversky Loss Function With Improved Attention U-Net for Lesion Segmentation“. In: *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. 2019, pp. 683–687. DOI: 10.1109/ISBI.2019.8759329.

[126] Rod Goodman, John W Miller, and P Smyth. „Objective functions for neural network classifier design“. In: *Proceedings. 1991 ieee international symposium on information theory*. IEEE. 1991, pp. 87–87.

[127] Adam J. Stewart et al. „TorchGeo: Deep Learning With Geospatial Data“. In: *ACM Transactions on Spatial Algorithms and Systems* (Dec. 2024). DOI: 10.1145/3707459. URL: https://doi.org/10.1145/3707459.

[128] Juan Terven et al. „A comprehensive survey of loss functions and metrics in deep learning“. In: *Artificial Intelligence Review* 58.7 (2025), p. 195.

[129] ESAOpenSR. *opensr-usecases: Super-Resolution Model Validator for Segmentation Tasks*. GitHub repository. 2025. URL: https://github.com/ESAOpenSR/opensr-usecases.

[130] Zeki Kuş and Musa Aydin. „MedSegBench: A comprehensive benchmark for medical image segmentation in diverse data modalities“. In: *Scientific Data* 11.1 (2024), p. 1283.