



Cross-Domain Image Registration with XFeat: Matching Infrared with RGB Imagery

Master Thesis

for the attainment of the Master's degree
"Master of Science (Continuing Education)", abbreviated "MSc (CE)"

submitted within the University Master Program for Further Education
"Geographical Information Science & Systems – UNIGIS MSc (CE)"

at the Department of Geoinformatics - Z_GIS,
Faculty of Digital and Analytical Sciences,
University of Salzburg

submitted by

B.Sc. Lilly Maria Kohaus

Supervisor:
Dr. Christian Neuwirth

Cologne, January 2026

Acknowledgements

I would like to express my sincere gratitude to my professional supervisor, Dr. Markus Brändle, for his invaluable support throughout this project. His expertise and insightful feedback were instrumental in shaping the direction of my work. I deeply appreciate his time, encouragement, and the opportunities to apply theoretical knowledge in a real-world context.

I would also like to extend my heartfelt thanks to my academic supervisor, Dr. Christian Neuwirth, for his continuous guidance, thoughtful suggestions, and constructive criticism. His academic insight and unwavering support helped me navigate the challenges of this research and improved the quality of the work presented here.

Table of Contents

1	Introduction	4
1.1	Evolution of Feature Matching	4
1.2	Neural Network Architectures and Matching Strategies	6
1.3	Related Work	8
1.4	Research Objective	25
2	Methods	27
2.1	The XFeat Architecture	27
2.2	The XFeat Repository	28
2.2.1	The Default Training Dataset	29
2.3	The Training Data	30
2.3.1	Creating The Ground Truth	32
2.4	Training XFeat with the Flight Dataset	35
2.5	Testing Configurations of XFeat	37
2.6	Metrics to Evaluate the Accuracy	38
3	Results	39
4	Discussion	46
4.1	Impact of Sample Size on Matching Performance	46
4.2	Impact of Dataset Rotation on Matching Performance	46
4.3	Performance of the Matching Methods	47
4.4	Limitations	47
5	Conclusion	48
5.1	Outlook	48
6	References	49

Abstract

Matching imagery of different modalities has been subject to many deep learning approaches. A variety of neural networks has been developed to master the task of cross- modality image matching that outperformed conventional, algorithmic methods. These networks differ in matching accuracy, computational complexity, speed and required resources for instance. Choosing the suitable neural network for a use case not only depends on the specific image matching task, but also on the restrictions of applicable hardware, run time and resources. The deep learning network XFeat (Accelerated Features) is characterized by its speed and resource efficiency, thus making it an suitable candidate for applications like image-based navigation and 3D reconstruction. However, it was not designed to match cross-modality image pairs, which often are the base for named tasks. This study examines if XFeat can be trained on a cross-modality dataset of (infrared) IR and RGB image pairs. It compares the different matching methods offered by the XFeat repository to each other and investigates the influence of dataset rotation and sample size. While, on a dataset without rotation variance, the mutual nearest neighbour matching approach performs best, dataset rotation does not make a difference for the coarse-to-fine matching approach. This thesis serves as initial step for further research on the performance of XFeat on cross-modality datasets.

1 Introduction

1.1 Evolution of Feature Matching

With the rapid development of remote sensing techniques, the variety of acquired data increased significantly. Remote sensing data of the same scene, captured by distinct sensor types, such as visible and infrared sensors, or at different times must be matched to fully utilize the provided information (Zhu et al., 2019). To address the challenges of image matching, research in the fields of intelligent processing, image based navigation, intelligent aircraft and targeting has intensified (Zhao et al., 2022).

The two basic concepts of image matching are area-based and feature-based approaches. Area-based approaches directly compare the input imagery by pixel similarity of colour values or intensity within a search window to compute correspondence (Huang et al., 2024). However, area-based methods are not robust to scaling, rotation or changes of illumination of the compared imagery. Feature-based matching, in contrast, relies on keypoint extraction and will be relevant for this thesis, as they make use of machine learning approaches (Verykokou & Ioannidis, 2025). These methods are widely used, as they are more robust against rotation and scaling (Zhu et al., 2019). The following section focuses on feature- based approaches.

A standard feature-based matching process is briefly summarized in three steps:

1. Keypoint detection
2. Feature description
3. Feature matching

Detection and Description are summarized as Feature extraction (Verykokou & Ioannidis, 2025).

The detection of features is performed independently in each image; the matching is based on comparing the descriptors that include information about position, size and the neighbourhood of the keypoint. Creating these distinct descriptor vectors is the most sophisticated task of the image matching process (Verykokou & Ioannidis, 2025).

For the longest time the image matching process was conducted using conventional algorithmic solutions, where detectors and descriptors were handcrafted. They are based on rules and mathematical models, such as gradient analysis for instance (Verykokou & Ioannidis, 2025). Examples for mentionable methods in this context are Difference of Gaussian (DoG) and

Laplacian of Gaussian (LoG) as keypoint detectors only; RIFT (Rotation-Invariant Feature Transform) for keypoint description only, SIFT (Scale-Invariant Feature Transform) and SURF (Speeded Up Robust Features) for detection and description (Zhao et al., 2022). The output of these algorithms is a feature descriptor, a n-dimensional vector, that describes each keypoint (Lazebnik, Schmid, & Ponce, 2005; Lowe, 2004). However, these classical feature descriptors have been found to be insufficient for matching imagery from different sensors, due to differences in spectral and geometric characteristics, resolution and lighting, including non-linear intensity variations (Aguilera et al., 2017; Wang et al., 2018).

In the past years, deep learning methods have proven to be sufficient alternatives to the traditional algorithmic solutions in the field of image registration. Deep learning networks learn the characteristics of features, which then perform extraction and matching of these keypoints, by analyzing given training data (Zhao et al., 2022). Two categories of deep learning approaches can be distinguished: detector-based methods and detector-free methods (Xu et al., 2024). These approaches were subject to many recent studies, that proved the reliability of neural networks (NN) (Wang et al., 2018). The features extracted by learning-based methods have shown to be more robust against geometric variations and changes in lighting and perspective for instance. The efficiency of this approach highly depends on the quality of the mentioned training data (Verykokou & Ioannidis, 2025). Another advantage is the possibility of a joint learning of detection and description, that allows optimization of the whole training process (Verykokou & Ioannidis, 2025). The network that pioneered combining these steps into a single network is SuperPoint (DeTone, Malisiewicz, & Rabinovich, 2018). Other deep-learning networks for joint detection and description are D2-Net (Dusmanu et al., 2019a), LF-Net (Local Feature Network) (Ono et al., 2018) and R2D2 (Revaud et al., 2019). Some approaches even combine handcrafted detection of keypoints with learning-based description; an example is HardNet, a learning-based descriptor trained with keypoints detected by DoG (Mishchuk et al., 2017).

Despite years of research and technological advancement image matching remains one of the core tasks of computer vision problems (Jin et al., 2021). Especially bridging the domain gap of heterogeneous remote sensing imagery remains a challenging problem (Liu et al., 2018).

1.2 Neural Network Architectures and Matching Strategies

Feature types are distinguished into global and local features. Global features describe an entire image; local features aim to describe keypoints or small regions of interest. While global features are useful for image comparison tasks, more distinct features are required for image matching (Hassaballah, Ali, & Alshazly, 2016).

Extracting local features has thus become essential part of modern feature-based learning-approaches.

An important characteristic of deep learning feature extraction is global context awareness. This describes the ability of a neural network to consider not only the keypoint -local feature- itself, but also its neighbourhood and spatial relationships, that are integrated in the descriptor (Deng, Birdal, & Ilic, 2018). While traditional solutions like SIFT and SURF extract only local features, meaning they neglect the spatial context of the described feature, deep learning approaches allow for extracting local features with global context awareness (Verykokou & Ioannidis, 2025).

This leads to another important distinction of learning-based approaches between purely convolutional networks and networks with an additional Transformer Frontend. Purely convolutional networks only operate in a local neighbourhood and do not capture global context information or dependencies on remote areas of the imagery. To overcome this restriction a self-attention mechanism, which is characteristic for Transformer architectures, amends the CNN to deliver global context for the descriptor (Bello et al., 2019). It allows to relate a feature to every other feature of the image. In this combined approach the CNN module extracts local features, which are then processed with the Transformer that considers the relations between distant features and can identify structural patterns across the imagery (Zhong & Jiang, 2025). This allows to combine local information and global context within the descriptor, producing more robust image matching results (Yang et al., 2025).

An overview of the feature-based matching methods is provided in Table 1.

Well-established handcrafted operators:
<ul style="list-style-type: none"> • Detectors: Moravec, Hessian, Förstner, SUSAN, Laplacian of Gaussian (LoG), Difference of Gaussian (DoG), MSER, FAST, AGAST; SIFT, SURF
<ul style="list-style-type: none"> • Descriptors: BRIEF, ORB, BRISK, FREAK, LATCH, SIFT, SURF
Well-established learning-based operators:
<ul style="list-style-type: none"> • Detectors: TaSK, TILDE, Quad-networks, Key.net
<ul style="list-style-type: none"> • Descriptors: HardNet, L2-Net, GeoDesc, SOSNet, HyNet, DenseGAP
<ul style="list-style-type: none"> • Detectors and Descriptors: SuperPoint, D2, R2D2, LFNNet, SILK, ALIKE, ALIKED, DISK
<ul style="list-style-type: none"> • Transformer-based, Detectors and Descriptors: SuperGlue, LightGlue, LofTR, MatchFormer

Table 1 Well-established methods for feature-based methods, separated into algorithmic (handcrafted) solutions and learning-based methods. Some of the learning-based methods combine the detection and description step. Excerpt taken from Verykokou and Ioannidis (2025)

Beside detecting and describing, the matching process is the last part of the image matching process. Here, three main types of matching are distinguished: sparse matching, dense matching and semi-dense Matching (Table 2).

Sparse Matching Just a few, selected keypoints are matched (e.g. SIFT, SuperPoint)

Dense Matching Every single Pixel is compared to the other image

Semi-dense Matching Only relevant and reliable regions in the image are compared – a compromise between sparse and dense

Table 2 Overview of the matching types

Many different architectures of neural networks have emerged, each with different performance characteristics depending on the input data. For imagery of every kind convolutional neural networks (CNN) in general have been found to perform best (Makosso, Almaktoof, & Abo-Al-Ez, 2025). CNN architectures are composed of three types of layers: convolutional layers, pooling layers and fully connected layers. This only provides a brief overview of the general composition of a CNN. The possible architectures and parametrizations of the individual layers are manifold (O'Shea & Nash, 2015).

The preprocessing of the data, feature scaling, the design of the applied neural network, training of the network and model evaluation for instance can be stipulated in a fixed sequence, a pipeline (Gerber & Pillay, 2022). The structure of this pipeline, including its starting and end point, depends on the use case and the available input data. It is crucial for the quality of the matching result (Wang et al., 2018).

Though, the learned features outperform handcrafted descriptors on numerous image registration tasks, they do not exhibit overwhelming superiority. Conventional detectors and descriptors still produce competitive results, as creating a sufficient end-to-end learning-based network remains challenging (Zhang et al., 2019). Hence, in studies on NN for image matching the proposed approaches usually compete not only against other learning-based methods, which are characterized by their robustness, but also against classical feature extraction methods, which have been state-of-the-art for decades (Yang et al., 2025).

1.3 Related Work

Cross-domain image matching has been subject to many studies, all of them approaching the problem differently. This chapter gives an overview of the evolution of architectures and overall frameworks for multi-modality matching. The most mentionable will be summarized to give an overview of the state-of-the-art architectures and approaches to enhance performance.

Convolutional Neural Networks superior to handcrafted approaches

The first study on matching heterogeneous image data using neural networks was conducted by Aguilera et al. (2016). Imagery of different spectral wavelengths- near-infrared (NIR) and visible range (VIS)- was used to train three different convolutional neural network architectures: 2-channel network, Siamese network and Pseudo-Siamese network (Figure 1). A Siamese network, also referred to as twin network, consist of at least two identical sub-networks with shared weights and parameters to compute similarity (de Rosa & Papa, 2022). Siamese networks play a role in various studies regarding matching of heterogenous image data. A pseudo-Siamese network is comparable in structure, but without shared parameters (Aguilera et al., 2016). These networks were already trained in the VIS domain and could be adjusted to process NIR image data. Aguilera et al. (2016) delivered evidence that CNN approaches were superior to algorithmic solutions, when matching imagery of cross-spectral wavelengths. The

study showed, that even untrained CNN capture better keypoints of objects than current handcrafted solutions. Additionally, these networks trained with NIR-VIS data also performed well with long-wave infrared (LWIR) datasets. The comparison of the three network architectures showed, that the 2-channel network model outperformed the Siamese network architectures (Aguilera et al., 2016, p. 6).

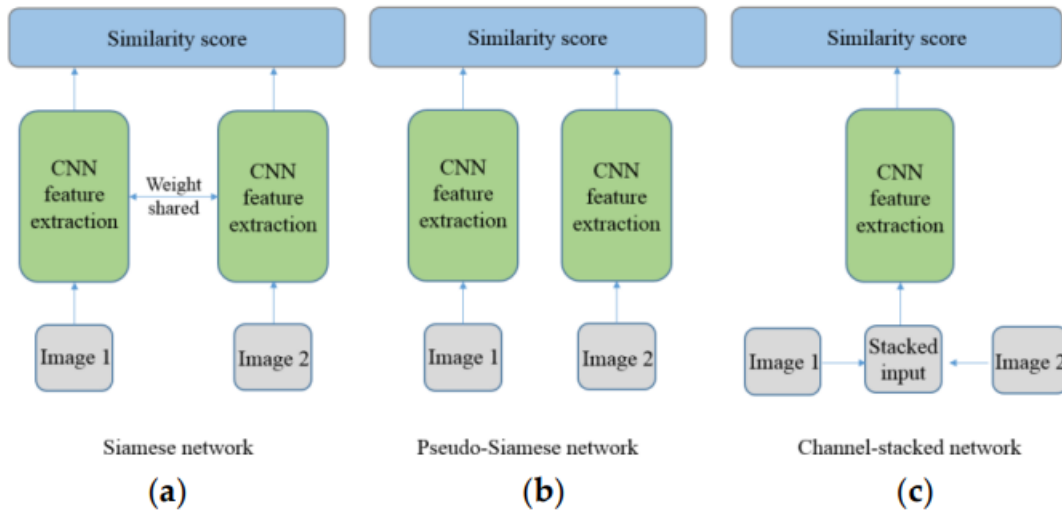


Figure 1 Comparison of the structures of a Siamese network (a), Pseudo-Siamese network (b) and Channel-stacked network (c) by Zhu et al. (2019) published under the Creative Commons Attribution (CC-BY) License (<http://creativecommons.org/licenses/by/4.0/>)

Adapting to spectral differences

Building on the findings of the previous research, another study by Aguilera et al. (2017) presented a new CNN architecture, called Q-Net. It was especially designed for learning local feature descriptors of different spectral bands. The architecture is based on the triplet network PN-Net by Balntas et al. (2016), which was constructed for mono-spectral image data. PN-Net consists of three CNN-towers, each taking a different image patch as input- two matching ones, and one non-matching patch. Aguilera et al. (2017) state, the key difference between matching cross- and mono-spectral imagery is that for every matching cross- spectral image pair, there are two possible non-matching pairs to serve as negative training samples (non-matching RGB patch and non-matching IR-patch). Thus Q-Net consist of four identical CNN-towers with different input patches (Figure 2).

The main idea of its functionality is later described as taking “[...] a positive pair and a negative pair consisting of optical and infrared photos as inputs to minimize the distance between positive samples and maximize the distance between negative samples” (Zhu et al., 2019, p. 3).

For the training and the evaluation of Q-Net a dataset of over 1 million VIS-NIR cross-spectral image pairs was used. The network was evaluated based on the false positive rate at 95% (FPR95).

The study showed that Q-Net is highly effective and outperforms PN-Net and other state-of-the-art methods when matching cross-spectral imagery. It also delivered more accurate results when matching VIS-NIR image pairs, while requiring less training data.

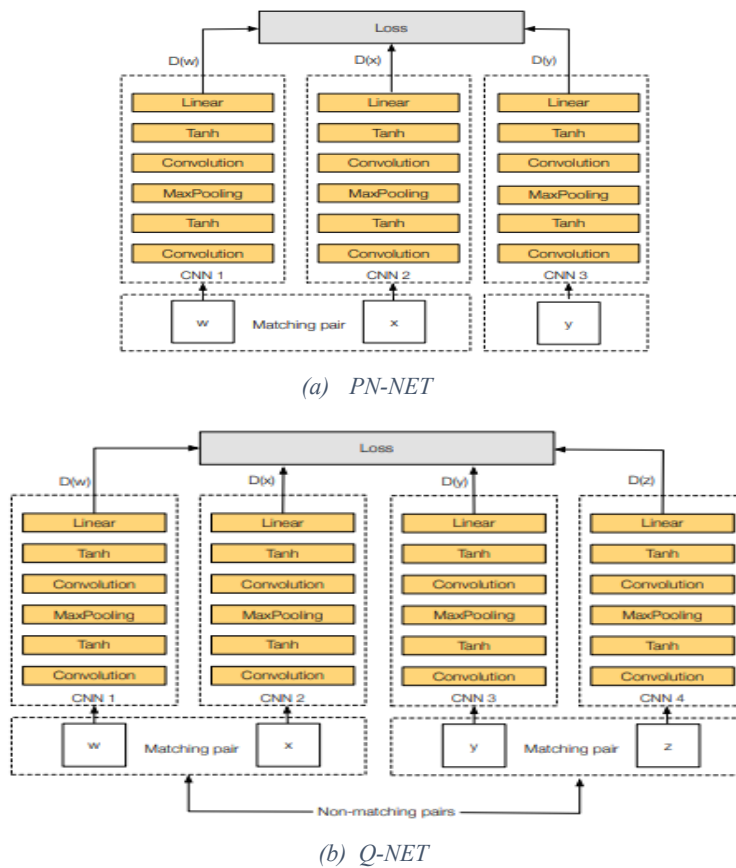


Figure 2 Comparison of PN-Net for matching mono- spectral patches and Q-Net, as adaption to cross-spectral image matching. Both architectures consist of duplicates of the same CNN. Published by Aguilera et al. (2017), under the Creative Commons Attribution (CC-BY) License (<http://creativecommons.org/licenses/by/4.0/>).

Combining a Siamese network with algorithmic approaches

While Q-Net tackles the problem of bridging cross-spectral differences, the study by He et al. (2018) addresses another, but related challenge. A framework for matching remote sensing imagery with complex background variations. Even though this thesis focuses on spectral and perspective differences, the non-linear intensity variation within the image dataset of He et al. (2018) used in the proposed study represents similar characteristics. The presented framework combines a Siamese convolutional neural network (SCNN) with traditional feature extraction methods, such as the Harris corner detector and Gaussian Pyramid Coupling Quadtree (Huang et al., 2024). The specific design of the SCNN trained in the study is based on the image characteristics that affect the matching performance, for instance: rotation and translation, non-linear geometric transformation, shadow and quality degradation (He et al., 2018, p. 23). The training data was inferred from Google Earth historical images and showed changes in land cover, illumination, scales and shadows.

The study showed that the proposed framework outperforms traditional methods, such as SIFT. It was compared to the deep learning approach by Zagoruyko and Komodakis (2015), that introduced the first approach to learn a similarity function directly from the raw input data, not requiring manually designed features.

The performance was evaluated using three metrics: number of correct matches (NCM), matching precision (MP) and root mean square error (RMSE) (He et al., 2018).

It outperformed the framework of Zagoruyko and Komodakis (2015), that failed on matching more complex remote sensing image pairs. Like in the study of Aguilera et al. (2017) the success of the approach by He et al. (2018) is inter alia explained by the domain-aware design.

Integrating an Autoencoder into a Siamese network

The approach of Liu et al. (2018) integrates an Autoencoder into a Siamese network. The shape appearing like the letter H gives this network its name H-Net. Although it performs well when matching cross-domain imagery, it does not perform effectively for descriptor retrieval. Thus, an enhanced version, H-Net++, is presented. It extracts more robust feature descriptors, which are less sensitive to changes in lighting and rendering styles for instance (Liu et al., 2018, p. 861).

H-Net consists of two Autoencoders as sub-networks, one for each domain, and a metric network. The Autoencoders have the same architecture, but do not share weights- making it a pseudo-Siamese network.

H-Net++ has the identical structure and Autoencoders, but instead of the metric network it is set up with a Euclidean distance constraint (Liu et al., 2018, p. 859). The former is a process of metric deep learning, to learn distances based on similarity in the embedding (feature descriptor) space. The Euclidian distance constraint restricts the straight-line distance between two embeddings to a specific range (Kaya & BİLge, 2019). The architectures are depicted in Figure 3.

While previous studies used real-world imagery with known perspective and spectral characteristics, Liu et al. (2018) introduced a fully synthetic cross-domain dataset. This data was purely generated for the purpose of this study by the CycleGAN (Zhu et al., 2017) algorithm. CycleGAN is a deep learning model that translates imagery from one domain to another, without requiring training data itself. It is useful when lacking matching pairs from both domains (Zhu et al., 2017). Two further datasets used in the study consists of camera imagery and rendered images from a 3D UAV model.

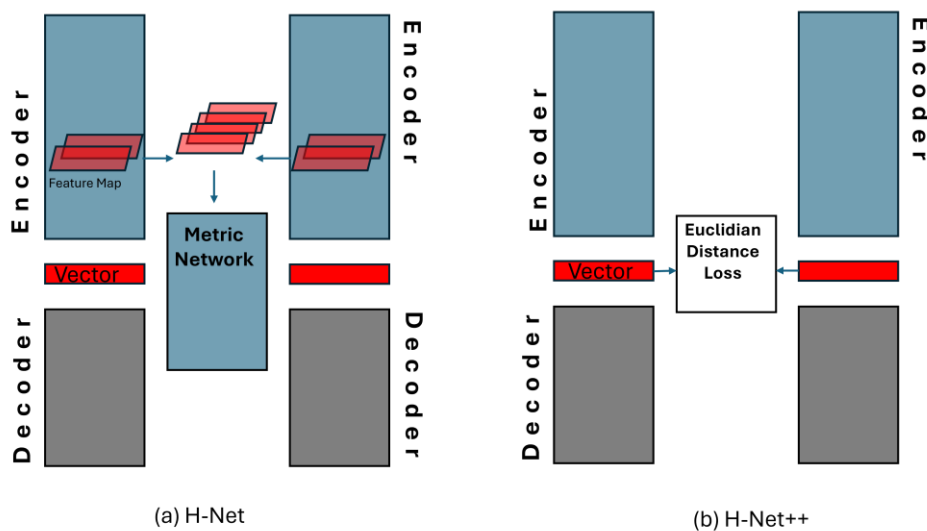


Figure 3 Schematic depiction of the differences of the architecture of H-Net and H-Net++. Two Autoencoders made up of an Encoder and Decoder form the basis of both networks. H-Net uses a Metric Network to learn distance function while H-Net++ employs an Euclidian Distance Loss to enforce a specific distance between similar and dissimilar embeddings. The information for his depiction are derived from the original figure by Liu et al. (2018).

A comparison of the TOP 1 and TOP 5 retrieval accuracy of H-Net with other Siamese network frameworks incorporating a metric network confirmed that H-Net delivers more accurate results- independent from the dimension of the feature descriptors extracted by the Autoencoders (Liu et al., 2018).

The same experiment was conducted with H-Net++, which was compared to similar Siamese networks without metric network. It significantly outperformed all other frameworks on cross-domain image matching. The authors announced further studies on H-Net++. Research on that remained without result.

Data augmentation and self-learning approach

The study of Wang et al. (2018) approached the difficulties of limited training data and the required manual annotation of large datasets when matching remote sensing imagery by automatically generating labelled training data. Although this study did not handle cross modality datasets, it introduced relevant approaches for remote sensing image registration and has since been referred to in research on cross modality matching.

The novelty of the approach is the data augmentation where transformed versions of the input training data are generated. The core idea is a “closed loop of information feedforward and feedback” (Wang et al., 2018, p. 163), which forms the basis of the self-learning approach. The neural network predicts matching labels and compares them to the correct labels that were initially determined by the model itself. It learns from the mistakes to improve.

First, a mapping function is learned from the original and transformed imagery. Second, image registration is performed by the trained neural network.

After applying a transformation to the unregistered input image, patches of the original image and its affine- transformation are identified. These patches are centred around keypoints that were extracted by conventional keypoint detection methods. Since the transformation is known, matching labels can be automatically generated (Wang et al., 2018). The process is depicted in Figure 4.

The imagery for the training dataset is collected from Radarsat, SPOT and Landsat.

To identify the best performing configuration of the NN the number of hidden layers and input patch size was varied.

The best performance was compared to seven state-of-the-art methods, mainly SIFT-variations. The only deep learning approach that was taken into consideration was Deep Learning SuperPoint Style (DLSS) (Ye et al., 2017).

The proposed framework of Wang et al. (2018) did not outperform all methods on every image data. Imagery used to create matching labels showing environmental changes over time, due to floodings or fires confused the proposed NN. It did perform best on very complex imagery with a lot of structural details (Wang et al., 2018, pp. 159-160).

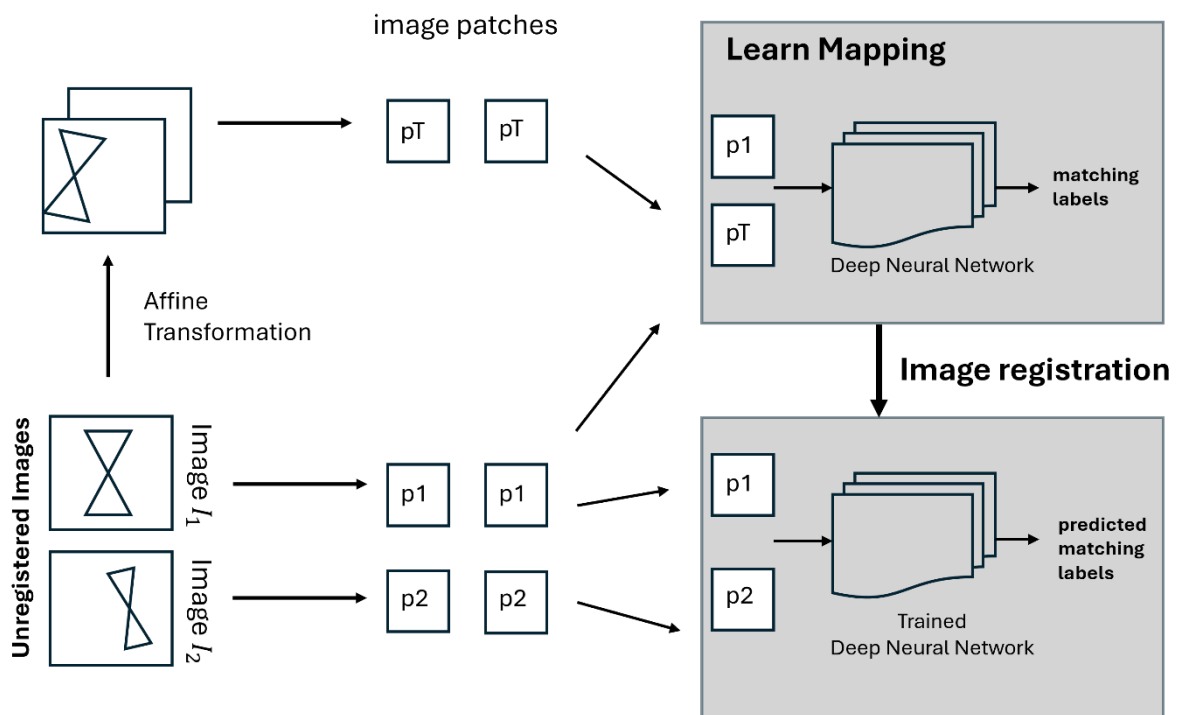


Figure 4 Schematic depiction of the proposed framework. The mapping function is learned from the patch pairs of the input image and its affine transformation and their matching-labels, which are derived from image I_1 . For the identifying correspondences the patches of Image I_1 and I_2 are matched with the learned mapping function (Wang et al., 2018). The graphic is based on the original depiction by Wang et al. (2018).

Combining a Siamese and Pseudo-Siamese network

While Wang et al. (2018) introduced a self-supervised learning pipeline, a study by En, Lechervy and Jurie (2018) combined the two common comparison methods of image patches from different modalities. Focusing either on common or the modality specific information in both patches. The approach combines a Siamese and a Pseudo-Siamese network to create a new network, called TS-Net (Figure 5). This results in a three- stream architecture: one shared stream from the Siamese-network (shared weights) and two independent streams from the Pseudo- Siamese network (unshared weights), that allow learning modality- specific features. It thus makes use of common and unique information of the cross-modal input patches (En, Lechervy, & Jurie, 2018). TS-Net predicts the similarity of cross-modal input patches independently in each sub-network; in an additional layer the outputs are combined to generate the final prediction (En, Lechervy, & Jurie, 2018).

TS-Net was tested on three different datasets, each containing imagery of two different sensors, for instance RGB-NIR patches. Additionally, a random affine transformation- rotation, translation and scaling- between the image pairs was applied, to create a perspective gap.

The conducted experiments showed, when to combine the information coming from the sub-networks to achieve the best performance. The options included “(a) after the feature extraction tower (b) after the first (c) second or (d) third layer of the metric network” (En, Lechervy, & Jurie, 2018, p. 4). The study found that a late fusion of the information works best. The results were compared to Siamese and Pseudo-Siamese networks. TS-Net outperformed these networks on all three datasets, demonstrating the effectiveness of combining shared weights and unshared weights for modality-specific processing for cross-modal patch matching (En, Lechervy, & Jurie, 2018).

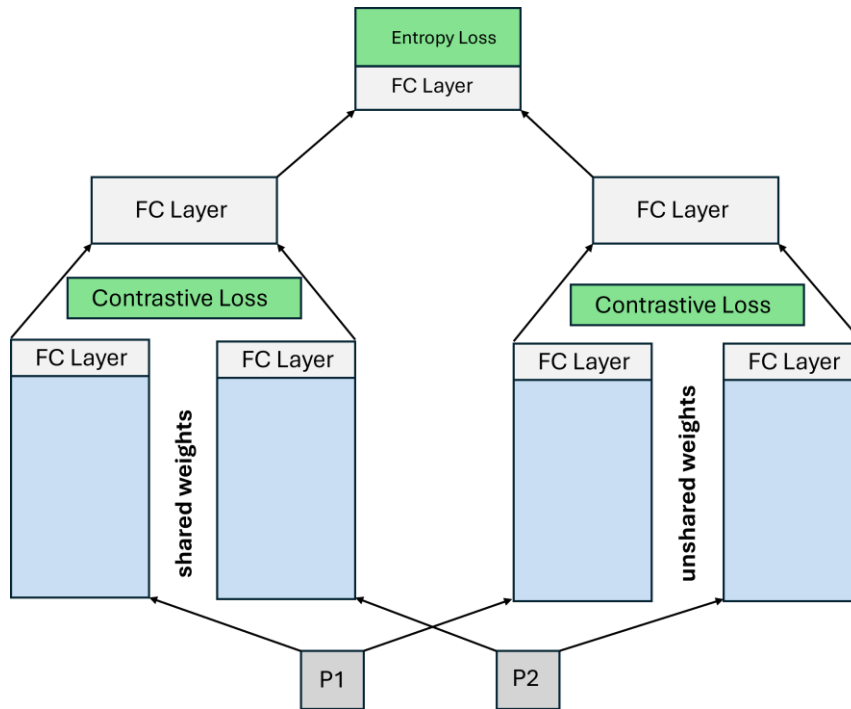


Figure 5 Schematic depiction of TS-Net. Each sub-network is made of two feature extraction towers and a learning module. The left sub-network is a Siamese-network with shared weights. The right network a pseudo-Siamese network with unshared weights and an additional contrastive loss (En, Lechervy, & Jurie, 2018). The graphic is based on the original depiction by En, Lechervy, & Jurie (2018).

Joint detection and matching in a single forward pass

The first study to introduce a CNN for jointly performing feature point detection and matching on optical and infrared imagery by using a single forward pass was done by Baruch and Keller (2018). Not requiring multiple passes through the network, it contrasts traditional approaches.

The neural network simultaneously learns to detect keypoints and match them, making detection and matching interdependent: both tasks are processed in one model and optimized together. The proposed network makes use of a similar approach as TS-Net and is described as “Hybrid CNN architecture consisting of both a Siamese sub-network and a dual-channel non-weight-sharing asymmetric sub-network” (Baruch & Keller, 2018, p. 2). The latter is required to handle the visual differences between the input imagery- each channel is specifically designed to process the input data of the different sensor. For each sub-network a different loss function is applied (Figure 6).

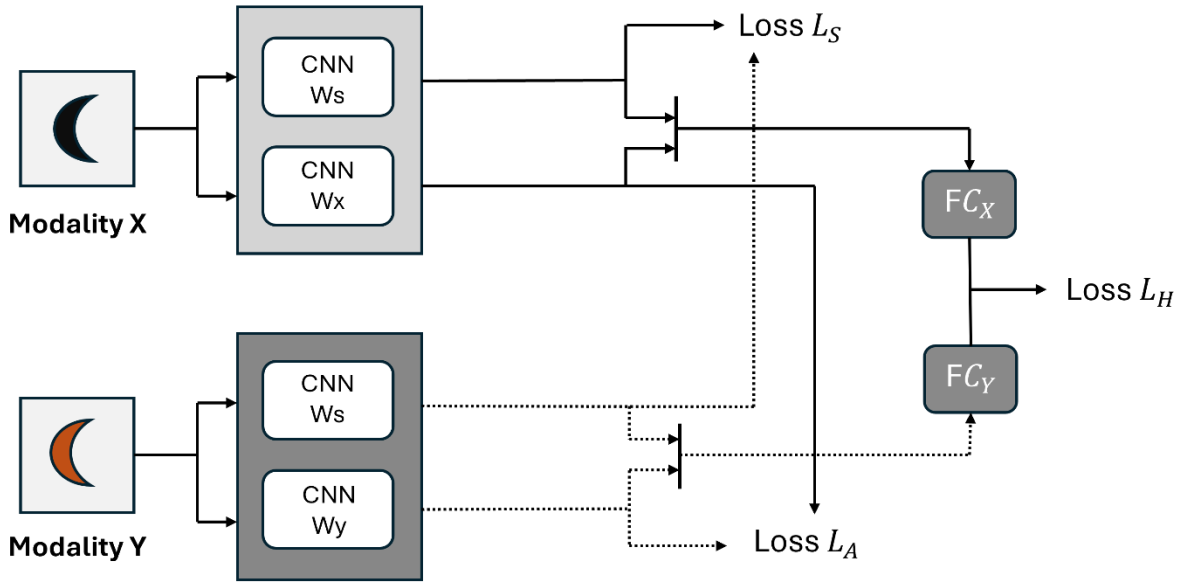


Figure 6 Schematic depiction of the proposed framework. The network consists of two sub-networks. The Siamese branch with shared weights W_s is trained by the Loss L_S . The asymmetric branch with non-shared weights W_x and W_y is trained by the Loss L_A . The output of both sub-network are combined through fully connected (FC) layers and result in the overall Loss L_H (Baruch & Keller, 2018). The graphic is based on the original depiction by Baruch and Keller (2018).

The newly introduced Hybrid CNN by Baruch and Keller (2018) was applied to three different datasets: visible and near-infrared (VIS-NIR) imagery, multispectral aerial imagery for vehicle detection and image pairs of faces and their corresponding artist sketches.

The training process included data augmentation by random rotations of 90° , as well as horizontal and vertical flipping of the input patches.

First, the performance of the proposed Hybrid CNN on the VIS-NIR dataset was compared to conventional algorithmic approaches. Like several deep learning methods proposed in previous studies, it outperformed these handcrafted descriptors significantly (Baruch & Keller, 2018).

Second, it was compared to the two Siamese based CNNs HardNet (Mishchuk et al., 2017) and L2-Net (Tian, Fan, & Wu, 2017), which were specifically trained for cross-domain matching on a VIS-NIR dataset. The proposed method by Baruch and Keller (2018) outperformed HardNet by 64% and L2-Net by 200%, quantified by the FPR95, thus setting a new benchmark for VIS-NIR patch matching (Baruch & Keller, 2018).

Furthermore, the feature point detection was evaluated. It was again compared to handcrafted detectors, like SIFT and the deep learning approach D2-Net (Dusmanu et al., 2019b), that was pretrained on RGB imagery with significant changes in lighting. D2-Net was tested in its

pretrained form, as well as trained on the multimodal datasets. The trained version performed worse than the pretrained version of D2-Net, which can be attributed to the smaller size of the multimodal dataset. SIFT outperformed in urban scenes but the Hybrid CNN was generally more robust across all modalities.

The study showed that the Hybrid CNN outperforms the state-of-the-art approaches in multimodal image matching.

Adapting stride value and loss function for the first generalized framework

While the study of Wang et al. (2018) approached the problem of training data limitation when matching remote sensing imagery, Zhang et al. (2019) focused on adapting a network to the characteristics of remote sensing imagery. As “*The previous approaches [Wang et al. (2018)] are all designed for single modal image matching*” (Zhang et al., 2019, p. 3), this study proposed a Siamese fully convolutional network (SFcNet), designed to process multimodal imagery. Its architecture is similar to L2-Net (Tian, Fan, & Wu, 2017) and HardNet (Mishchuk et al., 2017), but with a different stride value (number of pixels by which the convolutional filter moves across the image) on each convolutional layer. Also, a novel loss function was introduced. The study presented the first generalized framework for multimodal image matching, that allows for registration of various sensor types (Zhang et al., 2019).

The training datasets consisted of VIS-NIR (near-infrared), VIS- TIR (thermal infrared), VIS-SAR and VIS-Map (images taken from Google Maps) corresponding image pairs. The large spectral and temporal differences make the image matching process especially challenging. To generate a highly accurate training data set the matching precision was achieved in five steps, that sum up as follows (Zhang et al., 2019):

1. Coarse registration

The image pairs are coarsely registered by selecting four corresponding points (CP). An affine transformation is performed on the target image.

2. Feature Point Detection

Harris Corner Detection is applied to detect feature points in the reference image.

3. High-Confidence Matching

Local patches of 101 x 101 pixels for the reference image and 121 x 121 pixels for the target image are extracted around each feature point. The patches are matched, and outliers are removed.

4. Precise Registration

The whole imagery is matched with a piecewise linear (PL) transformation.

5. Generating more CP from imagery

By selecting Harris feature points in the reference image locate them in the target image.

The results showed that SFcNet performs excellent on VIS-NIR image pairs with a CMR (Correct Match Rate) > 94%, but not so well on the other cross-modal image pairs.

A study later by Zhu et al. (2019) comments on Zhang et al. (2019) “...SFcNet used the Siamese structure with shared weights for multimodal image matching. Due to the improper structure, it performed the worst and was 22.6% lower than ours on matching rate. In addition, the network was hard to train.” (Zhu et al., 2019, p. 7).

Channel-stacked CNN with template-based matching

Zhu et al. (2019) conducted a study on specifically matching RGB and infrared (IR) remote sensing imagery. It is claimed, that previously introduced approaches like Aguilera et al. (2016) and Baruch & Keller (2018) calculate a global similarity for the input image pair, instead of searching correspondences pixel-by-pixel. Zhu et al. (2019) refer to the study of Wang et al. (2018), as the only other approach to include this searching process, but finds it insufficient due to the heterogeneity of infrared and RGB imagery.

Zhu et al. (2019) also referred to the initially described study of Aguilera et al. (2017), where the superiority of channel stacked networks about Siamese networks was proven “Aguilera et al. showed that the channel-stacked network is superior to the Siamese networks in close-range visible and infrared image matching. The experiments of Baruch and Keller (2018) also showed that the channel-stacked networks are better than other variations of Siamese networks” (Zhu et al., 2019, p. 3).

Building on the findings of this study Zhu et al. (2019) developed an innovative channel-stacked CNN designed to match VIS-IR imagery. This approach tackles the three main problems with CNN based VIS-IR matching, that Zhu et al. (2019) identified as follows:

1. Limitation of convolutional layers, that affects the learning ability
2. No capability of pixel- or feature-wise correspondence search
3. Overfitting problem from the widely used cost functions

Commonly applied loss functions in CNN-based approaches are based on the single rule to maximize distances between negative (non-matching) samples and minimize distances between positive samples. This makes the model being overly confident in its calculations.

It is capable of dense correspondence search between VIS and IR images by replacing the feature-based through a template-based matching scheme.

Additionally, an augmented loss function “*which is the combination of the original cross entropy and an uniform distribution to make the network more general*” (Zhu et al., 2019, p. 5) enhances the learning stability and improves handling datasets from different modalities.

The training data consisted of five Landsat 8 image pairs, that were cropped to $620 \times 64 \times 64$ image patches.

The channel-stacked CNN was compared to the initially introduced 2-ch network (image-wise comparison) of Aguilera et al. (2016), the Siamese network SFcNet of Zhang et al. (2019) and a pseudo-siamese network. The applied metrics to evaluate the performance were the RMSE and the average matching rate (AMR), that was defined as “*the ratio between the number of correct matches and the number of all the reference points*” (Zhu et al., 2019, p. 7).

The channel-stacked CNN was also compared to conventional methods, such as SIFT, SURF and RIFT, which it clearly outperformed. The study highlighted the limitations of conventional approaches when matching imagery with large spectral differences.

The model of Zhu et al. (2019) outperformed the 2-ch network of Aguilera et al. (2016) measured by the AMR, with a 14.11% improvement for 1-pixel error and 5.35% for 2-pixel-error.

SFcNet had a 22.6% lower matching rate than the proposed network, confirming the superiority of channel-stacked networks as initially stated by Aguilera et al. (2016).

The results show that the template based approach of Zhu et al. (2019) is superior to recent CNN-based methods, that apply a feature-based strategy.

Reducing heterogeneity of the input data

While the above mentioned approaches focused on designing more complex feature extraction and description networks to bridge the differences in the heterogeneous image data, Zhao et al. (2022) tackled the problem by initially reducing the differences between the cross-domain input imagery. The proposed framework of the study matches satellite images and aerial images by initially converting the image data into the same domain, thus reducing the large feature differences. After this style transfer the imagery is matched by two networks D2-Net (Dusmanu et al., 2019a) and LoFTR (Sun et al., 2021).

The style transfer is conducted by using CycleGAN, transferring the satellite imagery into aerial imagery. The process of the correspondence mapping is depicted in Figure 7.

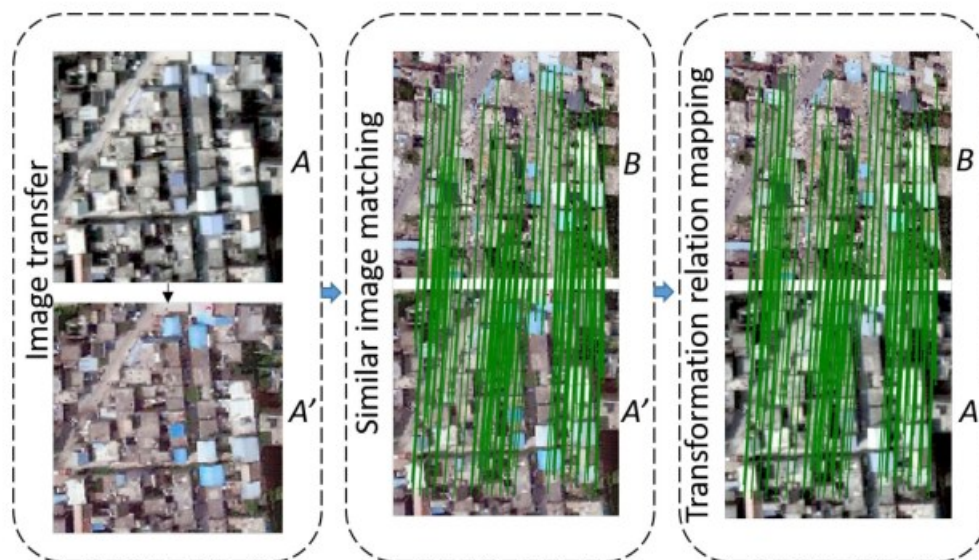


Figure 7 Depiction of the proposed style transfer with satellite image A and aerial image B . The generated aerial image A' is matched with the original aerial image B . The correspondence is then mapped to the original cross-modality imagery A and B . Figure by Zhao et al. (2022), published under the Creative Commons Attribution (CC-BY) License (<http://creativecommons.org/licenses/by/4.0/>).

While usually keypoint detectors initially create a feature descriptor to derive keypoints with different postprocessing methods, D2-NET combines detection and description, meaning it directly extracts the key features out of the feature descriptor (Zhao et al., 2022).

LoFTR does not include a keypoint detector. It extracts dense features pixel-by-pixel and directly computes correspondences between the input imagery.

The dataset consists of 1000 satellite image pairs and the style transferred correlates, using CycleGAN and SCycleGAN. The study proves that the style transfer increases the matching performance of both neural networks. It was also shown, that although the difference between

the aerial and satellite imagery was bridged, conventional methods like SIFT, SURF and ORB still fail to find correct matches; even after style transfer the image pairs are too heterogeneous. Zhao et al. (2022) proved that reducing domain differences before matching with CNN is an effective preprocessing step.

Mutual weighting strategy and recoupling

Another approach to bridge the domain gaps of cross-modal input imagery was presented by Deng and Ma (2022). The core of the problem is described as insufficient supervision of detection and inefficient coupling of detection and description. A mutual weighting strategy is introduced, where detection and description losses are weighted based on the feature reliability before recoupling them. This concept gives the network its name ReDFeat- Recoupling Detection and Description for Multimodal Feature Learning (Deng & Ma, 2022).

Furthermore, “Super Detector” is proposed to improve keypoint detection. It models the probability of a point being a keypoint and captures global context through a larger receptive field.

The network was trained with different kinds of cross-modal image pairs: VIS- NIR, VIS- IR and VIS- synthetic aperture radar (SAR).

The ReDFeat architecture is based on the R2D2 network (Revaud et al., 2019). It also couples detection and description but was not designed for multi modal matching. The R2D2 net consists of nine convolutional layers as the encoder, while ReDFeat uses the first six layers for modality specific processing to eliminate the variance of the different input imagery.

ReDFeat was compared to the conventional methods SIFT, RIFT, HN (Mishchuk et al., 2017) and R2D2 in terms of Feature Matching Performance, Image Registration Performance and Runtime. ReDFeat outperformed the other methods with a slightly increased runtime due to the complex computing operations of SuperDetector.

Mining Individual Features and Feature Relations enhanced by an Attention Mechanism

Yu et al. (2024) proposed RRL-Net (Relational Representation Learning Network) that not only considers the relation of the input features but also the intrinsic information of each input patch. An Autoencoder extracts features for each patch, while the Feature Interaction Learning (FIL) Module learns and describes relations between the different patch features. Unlike most previous approaches in cross-spectral image matching, the proposed framework makes use of an attention mechanism by applying lightweight multi-dimensional global-to-local attention (MGLA) module to extract local dependencies within global features (Yu et al., 2024).

The network consists of five components: an Encoder for global feature extraction, capturing local dependencies using the MGLA module; a Decoder to learn intrinsic features through self-supervised learning; a Feature Interaction Module (FIL), which extracts common and private features of image patches; Feature Aggregation to aggregate the private features from the FIL Module; and a Feature Metric to output a similarity score between patches (Yu et al., 2024).

The network was tested with four datasets: two VIS-NIR Patch Dataset, one of which was adapted with an affine transformation and two VIS-SAR (Synthetic Aperture Radar) datasets.

It was compared to 19 state-of-the-art-methods for VIS-NIR patch matching, including PN-Net, Q-Net, L2-Net, HardNet and conventional methods like SIFT. The evaluation based on the false positive rate at 95% recall (FPR 95). This metric measures how often a model matches unrelated image patches, when it correctly matches 95% of the true positive image pairs. RRL-Net more often fails to match positive pairs than it creates mismatches. Especially movements like grass in the wind and water and thus changing lighting conditions causing difficulties for the model. RRL-Net achieves the best mean performance based on the FPR95 of all compared networks and outperforms on four of eight subsets. It is faster and more efficient- having viewer parameters and smaller model size.

For the evaluation of the VIS- SAR patch matching, RRL-Net was compared to ten other networks. The proposed RRL-Net has the lowest FPR95, outperforming the compared methods quantitatively. It also has the highest visual robustness, meaning it can still correctly match image patches regardless of modal differences, lighting and weather changes or similarities of different objects.

Using single-modality training data for cross-modality matching

Acquiring cross-modality data that shows the same scene from the same perspective for identical timestamps can be challenging. While the previously presented approaches for cross-modal image matching required paired cross-modal training data to learn modality-invariant descriptors, a study by Liu et al. (2025) proposed a network for multimodal image matching using only single-modality training data. The main challenge for the modality-invariant feature learning network (MIFNet) *"...is learning modality-invariant features from such single-modality data"* (Liu et al., 2025, pp. 1-2). In the study MIFNet is combined with the XFeat architecture to improve the cross-modality matching performance of XFeat, which itself is not specifically designed for multi-modality image retrieval. The cross-modality matching performance of XFeat could be improved by the MIFNet, due to its strong modality invariance (Figure 10).

To learn modality-invariant features from single-modality data MIFNet applies a homographic transformation to each input image to generate a paired image. In another step keypoints and their corresponding feature descriptors are extracted from these image pairs, that are referred to as base features. Also, the ground truth relationship of the derived keypoints is determined to enable the self-supervised learning process, compute matching accuracy and handle unmatched keypoints.

To access the networks' ability to perform well on the cross-modal image matching task and evaluate the zero-shot capability of the proposed framework, MIFNet is trained and tested on two kinds of datasets:

1. Retinal Fundus Datasets

MeDAL-Retina Dataset (Nasser, Gupte, & Sethi, 2024) is used to train the network. The performance is evaluated on three cross-modal retinal datasets.

2. Remote Sensing Imagery

The network is trained on 1,200 color images of the SEN1-2 dataset (Schmitt, Hughes, & Zhu, 2018) and the VisDrone dataset (Zhu et al., 2021)

- The Optical-SAR dataset: with SAR images captured by the Chinese GaoFen-3 (GF-3) satellite
- The Optical-NIR Dataset: The VEDAI dataset (Ding et al., 2020), which is characterized by weather variations and object scale changes

MIFNet is only used with a base feature detector, which have been trained on keypoint extraction based on color fundus imagery. The datasets have been tested on state-of-the-art deep learning approaches alone and in combination with MIFNet, including the XFeat architecture. Across all datasets MIFNet enhances the matching performance, making single- modality approaches at least competitive if not superior to multi-modality approaches (Liu et al., 2025). Especially interesting for accessing XFeat on cross-modality datasets is the performance enhancement of XFeat if combined with MIFNet. The success registration rate improves by 43.6% on the CF-OCT dataset, 38.6% on the CF- FA dataset and 46.6% on the EMA- OCTA dataset. The performance on the OPT-NIR dataset could be improved by 8.5%.

The results are depicted in Figure 1 of the study by Liu et al. (2025).

1.4 Research Objective

The possible network architectures and overall pipeline structures are manifold. The most suitable approach depends on the input data, as well as the requirements for accuracy, processing time and given hardware. Options range from end-to-end deep learning approaches, combining learning-based with conventional methods or specifically processing the training data before the image registration process.

The data acquired for this thesis is obtained from different sensors: mid-wave infrared (MWIR) and aerial imagery (RGB). It shows scenery from different times and perspectives. The purpose of the matching process is the image based navigation and 3D reconstruction by identifying corresponding points in the image pairs. This requires real time matching and a hardware efficient architecture for resource- limited devices, that can provide highly accurate results.

A highly promising end-to-end deep learning approach that meets the requirements of the given use case is XFeat (Accelerated Features for Lightweight Image Matching). It employs a feature-based matching approach and is faster than state-of-the-art methods while maintaining the necessary accuracy (Potje et al., 2024). XFeat is designed to be efficient, fast and accurate, making it ideal for resource-constrained platforms. It is highly adaptable, hardware independent and can be easily modified and upgraded with elements of different architectures. An efficient additional component for the XFeat framework is LightGlue (Lindenberger, Sarlin, & Pollefeys, 2023), an improved version of the SuperGlue architecture (Sarlin et al., 2020). LightGlue is an attention-based network, that allows giving local features the mentioned global context. The GitHub repository with the relevant code of the XFeat architecture already contains critical elements of the LightGlue network. XFeat was not specifically designed to perform cross-modal image matching and has received very limited attention in the context of matching cross-modal image patches. It has been component of the presented neural network MIFNet proposed by Liu et al. (2025), that makes use of the XFeat single-modality detector. While this study showed that integrating MIFNet with the XFeat architecture improves its performance on cross-modality datasets, there remains limited research on XFeat's ability to match MWIR-RGB imagery. The study of Liu et al. (2025) delivered a solid base, but did not elaborate further on the different matching methods of the XFeat architecture.

The aim of this thesis is to evaluate the application potential of XFeat for matching images of the different modalities MWIR - RGB. The performance of two different matching methods-mutual nearest neighbour and coarse to fine- of the XFeat architecture will be compared, with respect to dataset rotation and sample size. The Code is publicly available at

https://github.com/verlab/accelerated_features/tree/main.

The methodology can be described as follows:

1. Create a labeled dataset/ ground truth
2. Create two datasets with different rotations
3. Train the network on the datasets
4. Perform image matching with the different matching methods
5. Evaluate the image matching performance based on a specific metric
6. Compare the results of the different methods and rotations

2 Methods

2.1 The XFeat Architecture

The XFeat Architecture is a purely convolutional network, characterized by its low computational costs, while maintaining state-of-the-art accuracy. This is mainly done by configuring the channel distribution- *“Our proposed strategy involves reducing the channel count in initial convolution layers as much as possible due to the high spatial resolution”* (Potje et al., 2024, p. 3). Usually, CNNs start with a higher number of channels in early layers and doubling the count of channels when the resolution is halved (Potje et al., 2024). XFeat starts with fewer channels in initial layers but triples their numbers in convolutional depth when the spatial resolution is halved. Another important characteristic is the separation of detection and description in the architecture. This makes XFeat more efficient while maintaining strong feature representation (Potje et al., 2024).

To keep computational costs low XFeat starts with fewer channels, when the resolution is high, and increases the number of channels when the image resolution is downsampled.

The XFeat backbone consists of two components: the Keypoint Head and the Descriptor Head.

The Keypoint Head extracts low-level features from the input image, representing it as 8x8 pixel grids. It computes a keypoint embedding with the keypoint locations within the 8 x8 pixel grid. A heatmap is generated, that depicts the probabilities of keypoints at certain locations.

The Descriptor Head extracts dense features to create a feature map. During this process the imagery is downsized to 1/32 of the original resolution. A reliability map predicts if the features can be matched confidently.

To make the matching process more efficient, only the most reliable regions of the images are selected, based on the reliability score derived from the reliability map (Potje et al., 2024).

The XFeat architecture is significantly different from the network architectures initially presented, which were specifically designed for processing cross-modality datasets.

The architecture of XFeat is depicted in Figure 3 of the study by Potje et al. (2024).

2.2 The XFeat Repository

The XFeat repository is composed of different modules and comes with a dataset to train the network, which will be examined below. To feed different training data into the network—especially when it is structured differently – can be laborious and requires understanding of each compound and their dependencies.

model.py	Implements the XFeat architecture. Defines the basic convolutional Layers in the Class BasicLayer and the composition of these in the Class XfeatModel, which extracts features, keypoints, and reliability maps from images.
xfeat.py	This is the inference component: this part of the XFeat Network applies an already trained model to new data. The Class XFeat extracts and matches the features of the unlabeled input imagery based on the learned metrics during the supervised training process. It includes three matching methods: <u>def match_lighterglue</u> : matches features using LighterGlue, a smaller version of LightGlue <u>def match_xfeat</u> : matches features based on mutual nearest neighbour (MNN) matching, a conventional algorithmic approach. <u>def match_xfeat_star</u> : a coarse-to-fine matching approach. First coarse features are matched; these are less precise but cover the whole image. Then the coarse match locations are refined using higher-resolution features.
interpolator.py	Extracts the feature vectors at the keypoint positions using interpolation and the feature map (output from the CNN).
lighterglue.py	A lighter version of the deep learning network LightGlue, with fewer parameters, making it faster (GitHub, 2024). Defines the class LighterGlue, that is applied for the matching process within the XFeat architecture. It takes as input the keypoints and descriptors extracted from two images by XFeat.
train.py	The main training script, that takes labelled training data to learn feature extraction and matching.

Table 3 The compounds of the XFeat repository

Another important element is the LighterGlue module, that can optionally be applied for the matching process. This is a more efficient and faster version of the transformer-based LightGlue network (Lindenberger, Sarlin, & Pollefeys, 2023), which was briefly described by Liu et al. (2025) “*LightGlue employs graph neural networks to aggregate positional information and descriptors through attention mechanisms*” (Liu et al., 2025, p. 2). LighterGlue was specifically trained for the XFeat architecture (GitHub, 2024). It allows to create global context for local features, enabling fast and semi-dense matches. It is not part of the XFeat backbone. It is applied after feature extraction by XFeat to perform matching. As LighterGlue is also a neural network, it must be trained to perform the matching process.

2.2.1 The Default Training Dataset

To understand for which use case XFeat was developed, the default dataset that is available in the repository is briefly depicted. The XFeat network is trained with the default datasets, that are derived from Megadepth (Li & Snavely, 2018) and synthetically warped COCO (Lin et al., 2014) images. The Megadepth dataset provides a large amount of training data, consisting of RGB imagery with corresponding depth maps. “*By using large amounts of diverse training data from photos taken around the world, we seek to learn to predict depth with high accuracy and generalizability*”(Li & Snavely, 2018, p. 2). It is generated from overlapping internet images, from which depth maps could be created. Megadepth is a single-modal dataset, with significant changes in perspective and illumination (Figure 8, Figure 9).



Figure 8 Image derived from the Megadepth dataset (GitHub, 2024). The imagery shows significant changes in lightening and less significant differences in perspective.

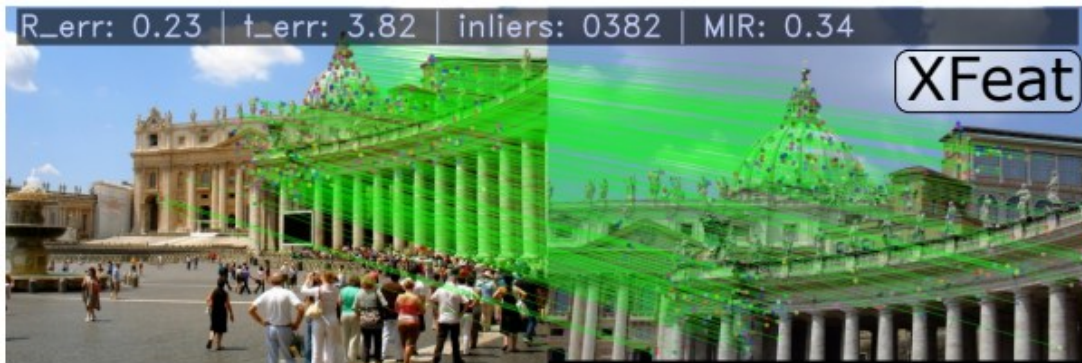


Figure 9 Image derived from the Megadepth dataset (GitHub, 2024). The picture shows correctly identified correspondences

2.3 The Training Data

All data used in this thesis was externally provided, including the imagery, the Ground Truth and the data feeder. The provided data was not changed. The processing of the data required to feed the imagery into the network was conducted in cooperation with the data provider. The processing steps are explained in this chapter.

Different than the default dataset, the training data used for this thesis consists of imagery of different spectra and perspective. While the RGB maps are orthorectified, the infrared imagery is taken from different angles.

The aerial imagery is acquired from the Bavarian Surveying Administration and have a maximum resolution of 0.4 m. The imagery is then cropped to fit the correlating MWIR image resulting in 2000 x 2000 pixel RGB patches. The center of each patch is the intersection point of the camera line of sight and the ground plane.

The infrared imagery is acquired by a manned ultralight aircraft and have a resolution of 640 x 512 pixel.

The imagery was taken in a total of eight flights between August 2023 and July 2024. All flights were conducted during daytime with varying weather and lightning conditions. The flight route is located between Augsburg and Geisenfeld. The landscape is dominated by agricultural use and arable land, which is characterized by its geometrical structure and clear separation. The grazing land is equally structured, but the colour nuances differ from arable land making it easy to distinguish between these two areas on the RGB as well as the infrared imagery. The imagery of the landscape also covers small forests areas and water bodies, such as rivers and lakes.

Urban infrastructures such as villages provide a significant contrast to the arable and grass land. Infrastructure elements such as roads and highways are sharply separated lines on the RGB and the infrared imagery.

The landscape is homogeneous and providing clearly distinguishable geometric patterns, recurring throughout the imagery. This makes it an ideal training dataset for the neural network to initially learn to extract and match keypoints on cross- domain imagery.

The dataset contains over 10.000 unique image pairs, after removing all nadir recordings of the infrared imagery.

Another variable is the orientation of the RGB maps. These are all north oriented, while the flight direction is arbitrary and varies across the dataset. Feeding the image pairs into the network would require it to learn not only the modality but also the rotation invariance.

To assess the influence of dataset rotation, another dataset is generated. It contains the identical imagery, but with a rotation applied. The RGB map is aligned with the flight direction, to minimize the rotation difference.

2.3.1 Creating The Ground Truth

To derive accurate ground-truth correspondences between the cross-domain and cross-view image pairs, a depth map is required. A depth map depicts the distance of each image point or pixel of the scene to the camera. If both camera parameters are known the corresponding points can be computed to generate a ground truth for the training process (Brown, Hua, & Winder, 2011).

To create a labelled training data set, corresponding points must be identified in the image pairs; this way the neural network can learn to match the raw, unaligned input data.

The integration of the new data takes place within the train method of the Class Trainer; the *get_corresponding_pts* method must be adapted to retrieve ground truth correspondences of the new dataset as a reference to train the neural network on our dataset.

First, the input imagery must be geometrically aligned.

Secondly, corresponding keypoints must be computed.

As the training data consists of multi modal imagery taken from different angles, the correct homography must be calculated to generate the ground truth. It describes how a pixel of the aerial imagery must be transformed to align with the corresponding pixel of the infrared imagery (Stolkin, Greig, & C, 2010).

To achieve this, the camera orientation and position of both sensor systems must be known to calculate a warp matrix. The warp matrix is defined by the relative pose of each camera and describes the projective transformation that must be applied to each pixel of the RGB map to align with the infrared image (Dlesk, Vach, & Pavelka, 2021). First a point cloud is generated from the aerial image using the depth map and the intrinsic camera matrix. These 3D points are then transformed into the coordinate system of the infrared imagery using the relative pose between both cameras (Zhang et al., 2021).

In a final step these transformed 3D points are projected onto the image plane of the infrared imagery.

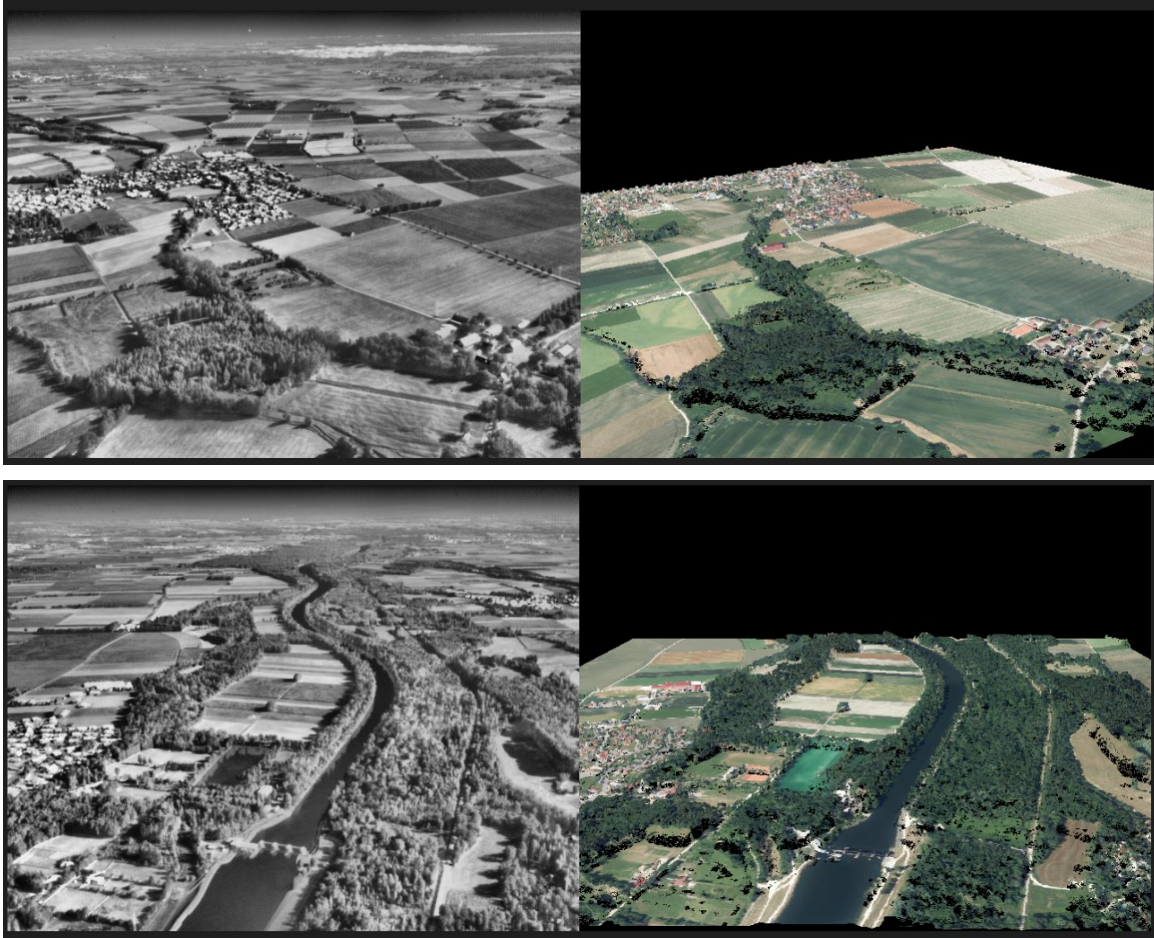


Figure 10 Example picture of the training data. Left: IR imagery, right: corresponding RGB imagery transformed into the IR image plane

The transformation is test wise applied to the RGB image. To verify the calculation, the result of the projection can be plotted (Figure 10).

Now the corresponding points can be calculated, by determining where each Pixel of the RGB image is located in the infrared image after applying the projection matrix. Then, die distance between every pixel of the infrared image and the projected pixels of the aerial image is calculated. Pairings with the shortest distance are the correspondences. The result is visualized in Figure 11.

This set of pixel-wise correspondences between the two images is the ground truth for the training process and serves as labelled data. It allows the neural network to learn how to predict corresponding points across RGB and infrared imagery.



Figure 11 Two image pairs from the training dataset. The lines mark the correspondences calculated between the MWIR and the RGB image

This way the neural network can be supervised when learning which features correspond in each imagery; after the training process the network no longer requires the correspondences. The purpose of the training process of the network is to create identical descriptors for features that correspond to the same physical location in the input imagery, as in the matching process the network only compares similarities of the descriptors for two features. The objective is to generate identical descriptors that represent the same geometric content. For this thesis, the code was provided externally.

2.4 Training XFeat with the Flight Dataset

The training process was conducted with 90% of the dataset. The remaining 10% were used as the evaluation dataset to assess the networks' ability to extract and match the input imagery.

The Software used to train and test the network was Visual Studio Code. The XFeat repository was cloned from GitHub. The processed data and the data feeder were accessed via a SSH Server.

The network was trained separately on both datasets, the aligned and unaligned imagery.

The training was conducted with different samples sizes of {5.000, 10.000, 15.000, 20.000, 25.000, 30.000, 35.000, 40.000, 45.000, 50.000} image pairs for each dataset. The sample sizes were generated by different permutations of the unique image pairs. A randomizer feeds the imagery into the network in a different order each training cycle and ensures the permutations are diverse enough. This allows to evaluate the matching performance in reference to the sample size and determine when overfitting might occur. The sample sizes consist of the same set of imagery for training on the differently rotated datasets, to allow for comparability of the matching results across the datasets.

The outcome of the training process are the weights specific for every sample size and dataset. The generated weights will be loaded in the network when testing the performance on the remaining imagery.

Different than the algorithmic approaches `match_xfeat` and `match_xfeat_star`, the `match_lighterglue` method uses deep learning to match the extracted embeddings. Thus, this network must be additionally trained on the flight dataset to learn how to match the embeddings. This will not be part of this thesis.

In the forward propagation process the training input is loaded into the neural network. In the backpropagation process the output of the training process is compared to the given target output- the labelled training data. This comparison of the predicted output and the target output is expressed with the loss function. The loss function measures the ability of the network to model the training data. The aim of the training process is to minimize the loss between the predicted and the target output (Elharrouss et al., 2025). To do so, the hyperparameters of the neural network can be adjusted. Hyperparameters are settings, that are independent from the

input data and cannot be learned. They are set in advance to define the training process.

Examples for Hyperparameters are:

- Learning rate
- Batch size – number of samples in each epoch
- Number of epochs – how often does the NN process the dataset
- Dropout rate – how many data is randomly ignored to avoid overfitting
- Number of hidden layers – depth of the NN
- Number of neurons per layer – width of the layer

(Ebadi, Kaur, & Liu, 2025).

In this thesis, the hyperparameters of the XFeat network were not being adapted and are identical across the different configurations. The reasons for this are that on the one hand, the main differences are expected by changing architectural modules and adaptations in the input data; on the other hand, the additional analysis of hyperparameter optimization would exceed the circumference of the thesis.

2.5 Testing Configurations of XFeat

The weights generated in the training process were used for evaluating the performance on the remaining 10% of the flight dataset. The matching results for the network trained on the different sample sizes were compared to each other. To do so, the weights according to the sample size were loaded into the network.

The following methods were compared to each other:

1. **match_xfeat with unaligned image pairs**

A mutual nearest neighbour approach. The Brute Force Matcher compares every descriptor of the infrared image with every descriptor of the RGB map. The imagery is fed into the network with different orientations. The RGB map is oriented to the North and the MWIR image in an arbitrary flight direction. This makes the matching process more difficult but allows more flexibility with the input data.

1.1. **match_xfeat with aligned image pairs**

The same approach, but with the RGB map oriented in flight direction.

2. **match_xfeat_star with unaligned image pairs**

Initially extracts coarse features across the image as approximate correspondences and then refines matches (Li et al., 2023). The RGB map is oriented to the North and the MWIR image in an arbitrary flight direction.

2.1. **match_xfeat_star with aligned image pairs**

The same approach, but with the RGB map oriented in flight direction.

3. **Zero-shot testing of match_lighterglue with unaligned image pairs**

A deep learning approach that matches the extracted features by learning the correct correspondences. As the lighterglue network has not been trained on the flight dataset, it does not know about the domain and perspective specific differences. The weights used in this approach are the default weights generated on the Megadepth dataset, making this a zero-shot approach. Only the influence of the dataset rotation will be assessed.

3.1. **Zero-shot match_lighterglue with aligned image pairs**

The same approach, but with the RGB map oriented in flight direction.

2.6 Metrics to Evaluate the Accuracy

Examples for common metrics to evaluate a network’s matching performance are the root mean square error (RMSE) and success registration rate (SRR). These were applied for evaluating the performance of MIFNet leveraging XFeat (Liu et al., 2025). The SRR is defined as the proportion of successfully registered images to the total number of images (Liu et al., 2025). The RMSE measures the average squared difference of the distance between the matching points. Both metrics refer to the whole image and express how well an image pair matches, considering all pixels or features (L. Li et al., 2022).

In this thesis the focus lies on the general ability of XFeat to find correspondences in the cross-modality dataset.

A metric that allows to quantify the number of matches is the mean matching accuracy (MMA).

A study applying this metric was conducted by Li et al. (2025), where the MMA is defined as “*the average percentage of correctly matched pixels across different threshold values*” (Li et al., 2025, p. 10). The MMA of the proposed network is calculated for the error thresholds from 1 to 10px.

For evaluating the results of this thesis, the MMA was used. This metric measures the accordance between the prediction of a model and the actual result. It is defined as “*The ratio of the number of correctly matching keypoint pairs to the number of pre-matched keypoint pairs in the two images*” (J. Li et al., 2022, p. 2535).

This error threshold defines the maximum distance between a pair of keypoints for it to be seen as a correct correspondence (J. Li et al., 2022). The error thresholds for the evaluation of the results of this thesis are {5, 10, 20} pixel. Though, common error thresholds range from 1 to 10 px, a greater error threshold is considered in this thesis. As XFeat is not made to match a cross-modality imagery, in this first approach the general ability to find correspondences is tested under more relaxed requirements. Also, the difference between the sample sizes are more distinct for greater error thresholds.

The MMA is calculated for every image pair per threshold. It must be averaged across the images, to receive one value per sample size and threshold.

3 Results

The results were generated and evaluated on the separate dataset, that has not been used for the training process. Due to confidentiality reasons the absolute values of the MMA will not be released. The results of the matching process are depicted in graphics that show the relative MMA based on the best performance, which is the reference normalized to 1. This shows how well a sample size performs in direct comparison to the best performance. The x- axis represents the error threshold, the y-axis represents the Ratio of the MMA.

1. Match_xfeat

1.1. match_xfeat with unaligned image pairs

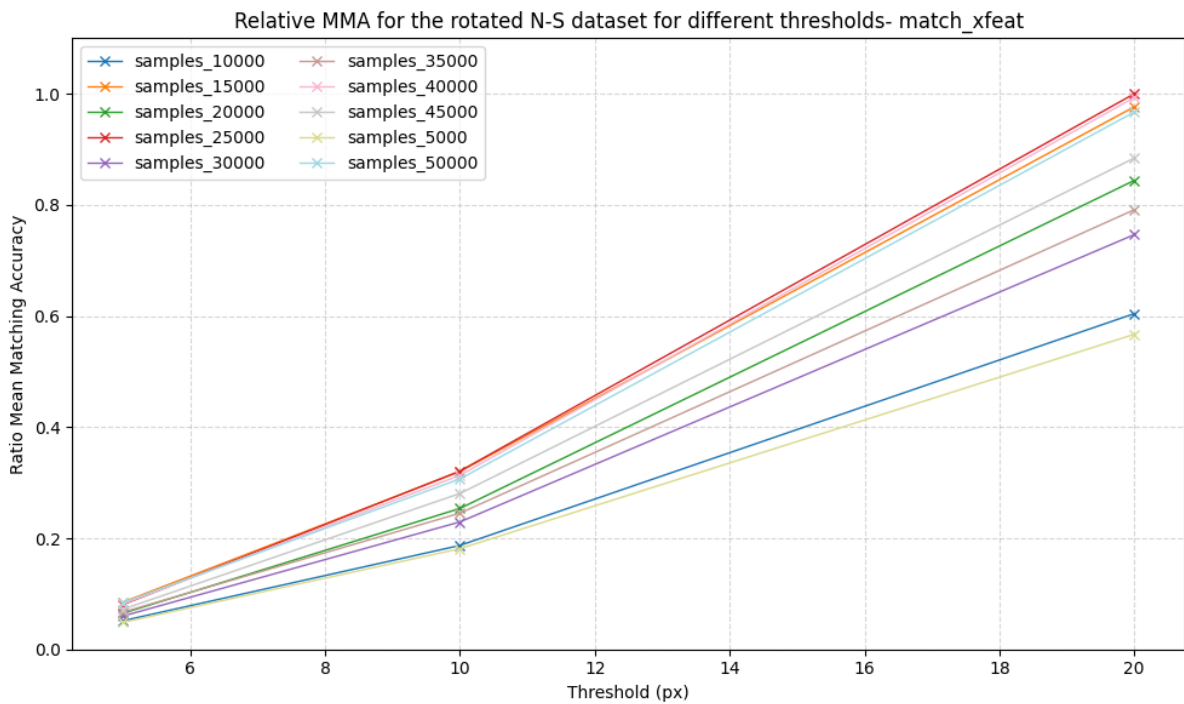


Figure 12 Comparison of the matching accuracy using training results of different sample sizes. The graphic illustrates relative accuracies based on the best matching result.

Figure 12 depicts the matching performance for the mutual nearest neighbour method match_xfeat for the unaligned dataset. Each graph represents a different sample size.

The best results were achieved with 25.000, 40.000, 15.000 and 50.000 samples across all error thresholds. A sample size of 25.000 slightly outperforms greater sample size of 50.000 for a threshold of 20px. 20.000 samples deliver slightly better results than 35.000 and 30.000 samples. Training the network with 5.000 and 10.000 samples delivered relatively inaccurate results. The graphs are all parallel, the influence of the sample size is the same across all

thresholds. For a small error threshold, the difference in performance for different sample sizes is insignificant. With greater error thresholds the influence of the sample size becomes more visible. The graphs show a steeper increase in the 10 to 20 range of the error threshold, compared to the 5 to 10 range. The MMA improves more significantly for greater thresholds.

1.2. match_xfeat with aligned image pairs

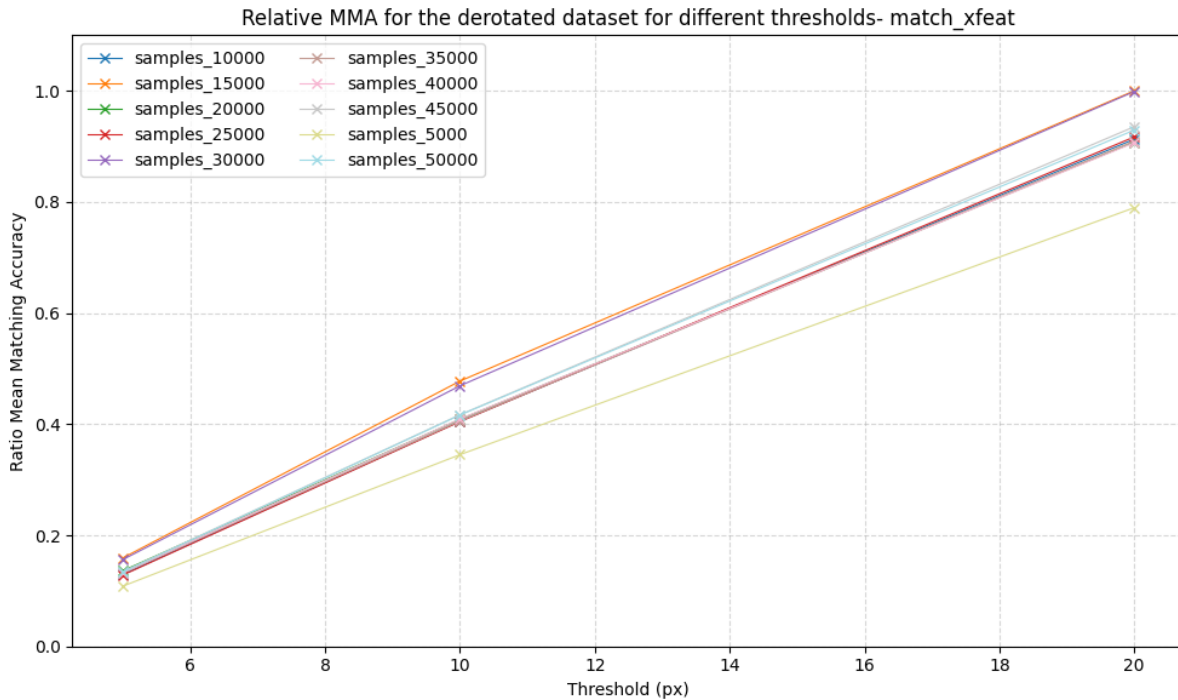


Figure 13 Comparison of the matching accuracy using training results of different sample sizes. The graphic illustrates relative accuracies based on the best matching result.

Figure 13 depicts the matching performance for the mutual nearest neighbour method `match_xfeat` for the aligned dataset. Each graph represents a different sample size.

The best performance is achieved for 15.000 and 30.000 samples, standing out clearly across all thresholds. Training the network with only 5.000 samples delivers relatively inaccurate results, standing out way below. Excluding the best and the worst results, all performances across the different sample sizes are very similar- the differences are insignificant. The performances are in general slightly better, all above 80% of the best performance compared to the unaligned dataset for the highest error threshold. Different than the results of the unaligned dataset, the graphs show a steeper increase in the 5 to 10 range of the error threshold, compared to the 10 to 20 range. The MMA improves more significantly for lower thresholds.

1.3. Comparison of the results of aligned and unaligned image pairs

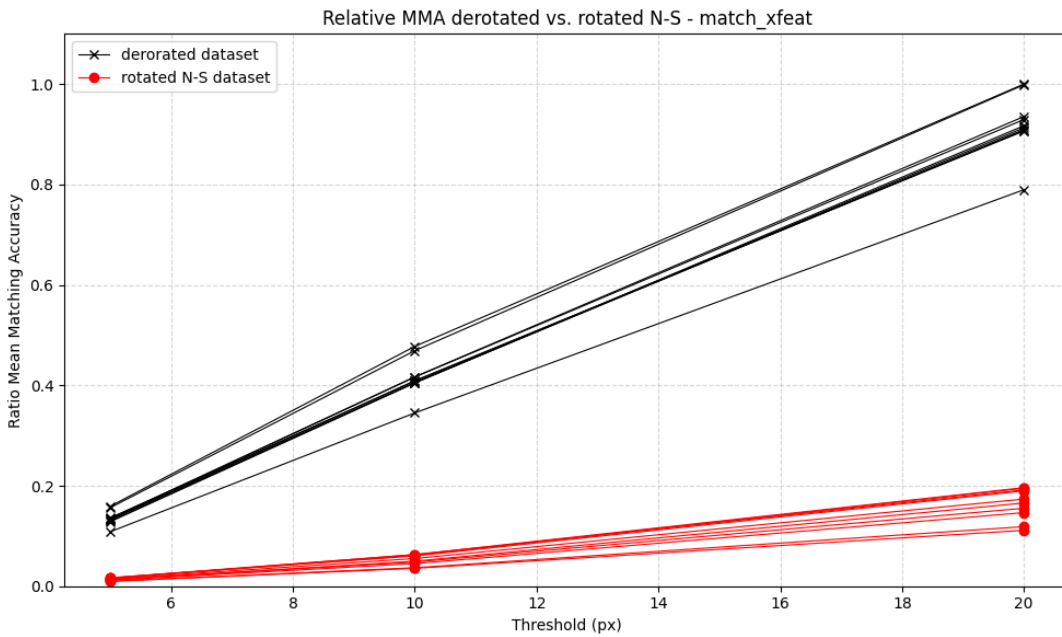


Figure 14 Comparison of all results for the derotated and rotated dataset

Figure 14 compares the performance of the derotated dataset (black) to the rotated dataset (red) across all sample sizes for the match_xfeat method. It visualizes the significant performance difference between aligned and unaligned input data. The derotated dataset performs significantly better than the rotated dataset across all sample sizes. The rotated dataset delivers a maximum of 20% of the MMA of the derotated dataset.

2. match_xfeat_star

2.1. match_xfeat_star with unaligned image pairs

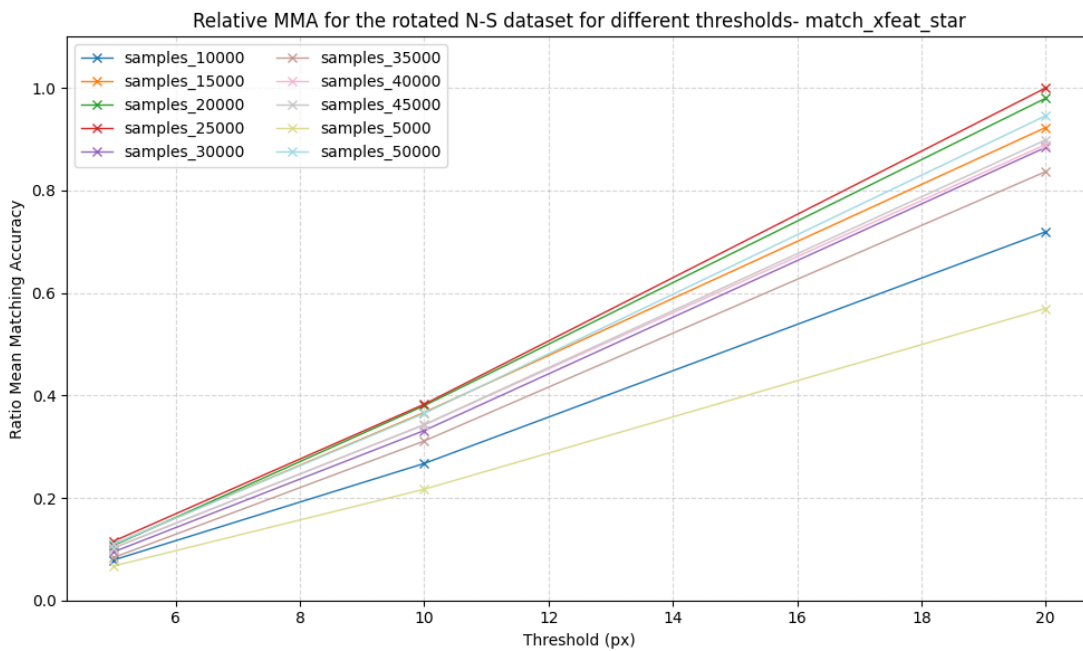


Figure 15 Comparison of the matching accuracy using training results of different sample sizes. The graphic illustrates relative accuracies based on the best matching result.

Figure 15 depicts the matching performance for the coarse to fine method `match_xfeat_star` for the unaligned dataset. Each graph represents a different sample size.

The best performance is achieved for 25.000 samples across all thresholds, closely followed by 20.000 samples. Weights generated with 30.000 samples lead to better performance than 35.000 samples. Training the network with only 5.000 and 10.000 samples delivers the most inaccurate results. Like the results of `match_xfeat` with the unaligned dataset, the graphs show a steeper increase in the 10 to 20 range of the error threshold, compared to the 5 to 10 range. The MMA improves more significantly for greater thresholds. With a higher error threshold, the performance differences across the sample sizes become more significant.

2.2. `match_xfeat_star` with aligned image pairs

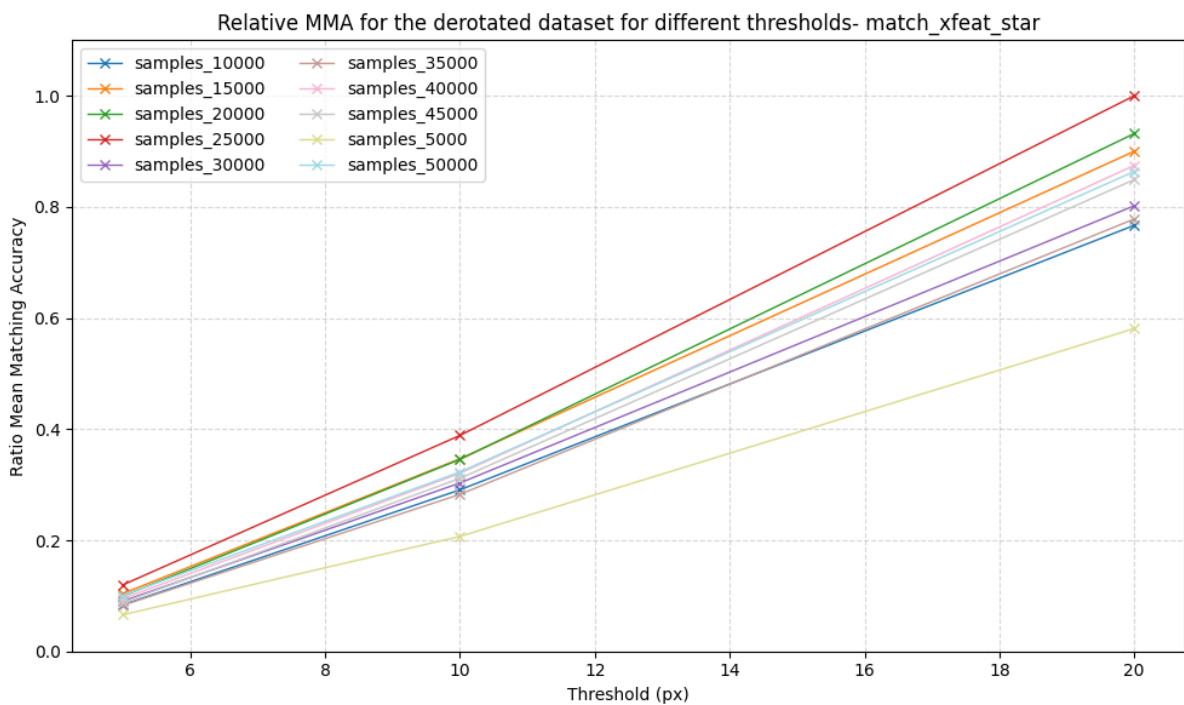


Figure 16 Comparison of the matching accuracy using training results of different sample sizes. The graphic illustrates relative accuracies based on the best matching result.

Figure 16 depicts the matching performance for the coarse to fine method `match_xfeat_star` for the aligned dataset. Each graph represents a different sample size.

The performance for the aligned dataset is similar to the performance of the unaligned dataset. The best matching results are achieved with 25.000 samples. A sample size of only 5.000 delivers the worst results. Across all thresholds greater sample sizes do not lead to better performance. The distance of the best performing sample size to the rest is higher compared to the unaligned dataset.

2.3. Comparison of the results of aligned and unaligned image pairs

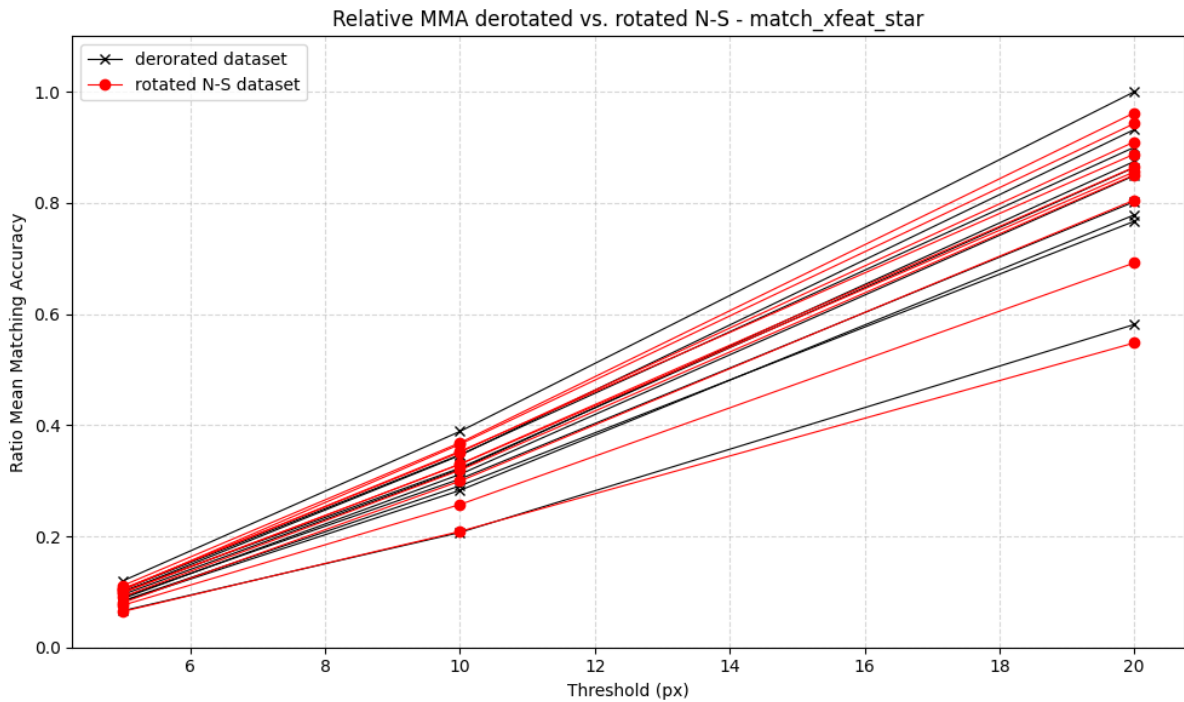


Figure 17 Comparison of all results for the derotated and rotated dataset

Figure 17 compares the performance of the derotated dataset (black) to the rotated dataset (red) across all sample sizes for the coarse to fine method `match_xfeat` method. It visualizes how aligning the dataset barely influences the matching performance.

For the `match_xfeat_star` method the difference between the rotated and derotated dataset is not significant. The relative MMA is almost the same across all datasets. The graphs behave similar across all thresholds.

3. Comparison of all configurations

Figure 22 compares both methods for the aligned and unaligned dataset.

The best matching accuracy is achieved with the `match_xfeat` matcher and aligning the RGB map in flight direction, significantly standing out from the other methods. The next best result, achieved with the `match_xfeat_star` matcher only reaches approx. 24% of the best performance for a high error threshold. Feeding unaligned imagery into the network and matching it with `match_xfeat` delivers the most inaccurate results. The performance of the aligned and unaligned dataset for `match_xfeat_star` is identical across all thresholds. The graphs behave similar, while the graph representing `match_xfeat` with the aligned dataset increases steeper across all thresholds.

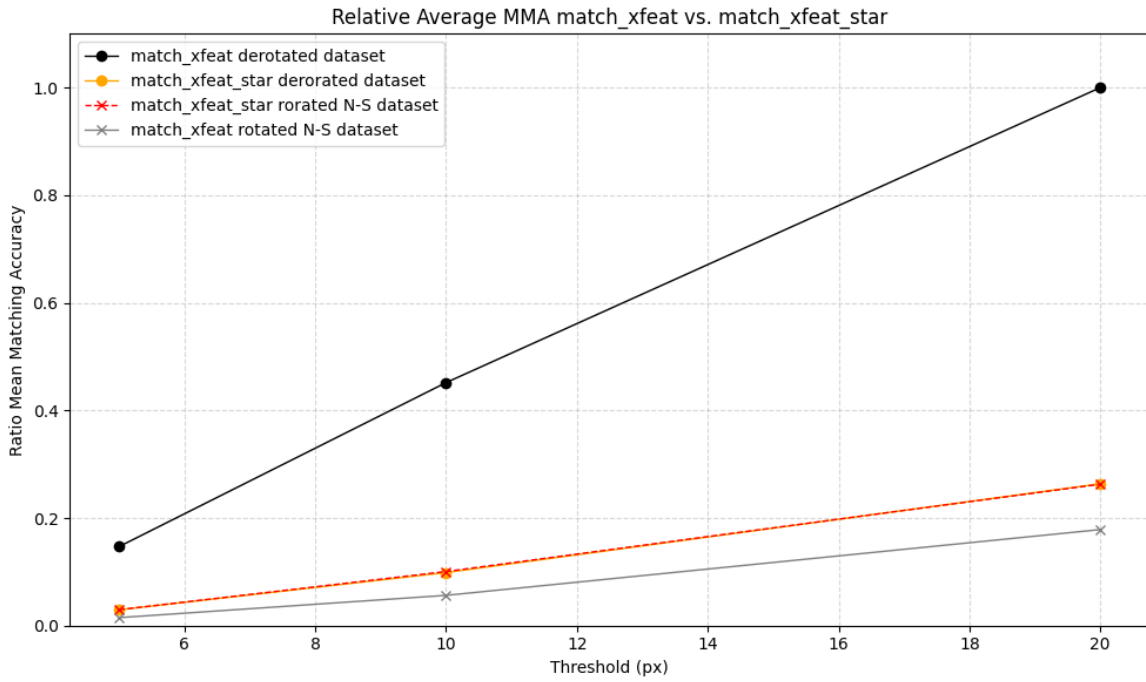


Figure 18 Comparison of all datasets. The graph of each dataset represents the average of the MMA across all sample sizes.

4. Match_lighterglue untrained on the flight dataset

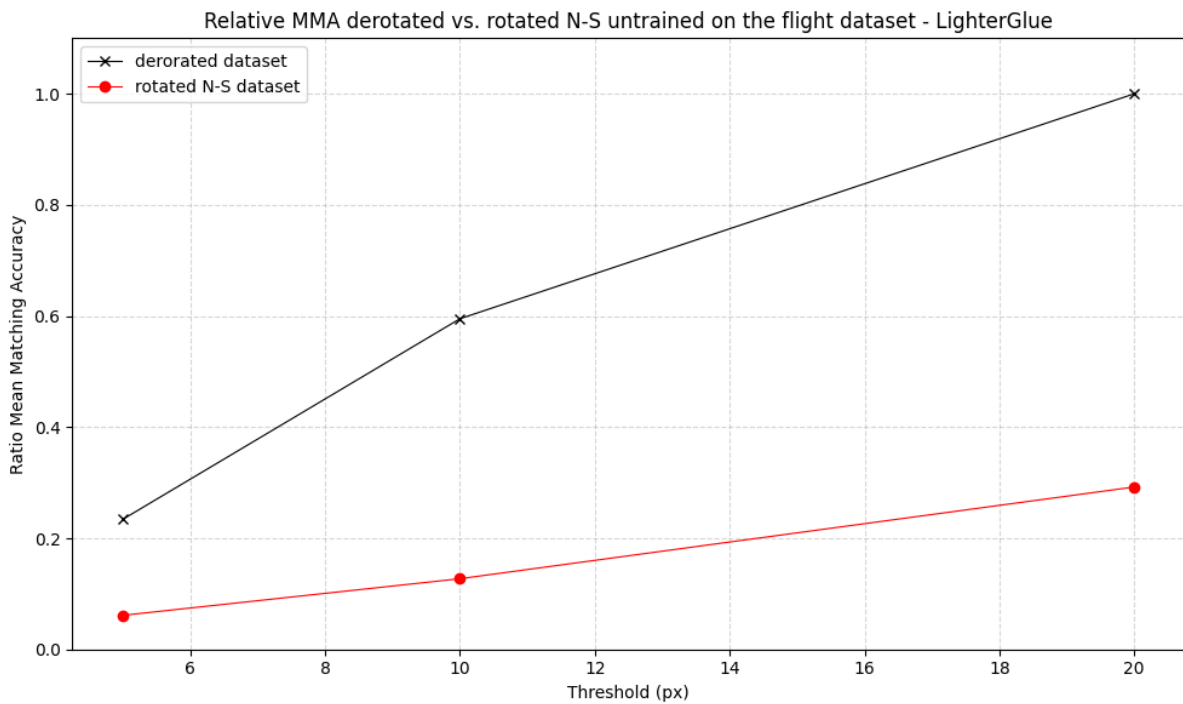


Figure 19 Matching results for LighterGlue. The weights applied are the default weights, trained on the MegaDepth dataset.

Figure 19 compares the performance of the match_lighterglue method for the aligned and unaligned dataset.

The performance of the derotated dataset is better across all thresholds. The difference increases with higher thresholds.

The matching result for the unrotated dataset reaches a maximum of 30% of the accuracy of the derotated dataset for higher thresholds. The MMA for the rotated dataset increases proportional to the increasing error threshold, while the derotated dataset shows a steeper increase in the 5 to 10 pixel range of the error threshold.

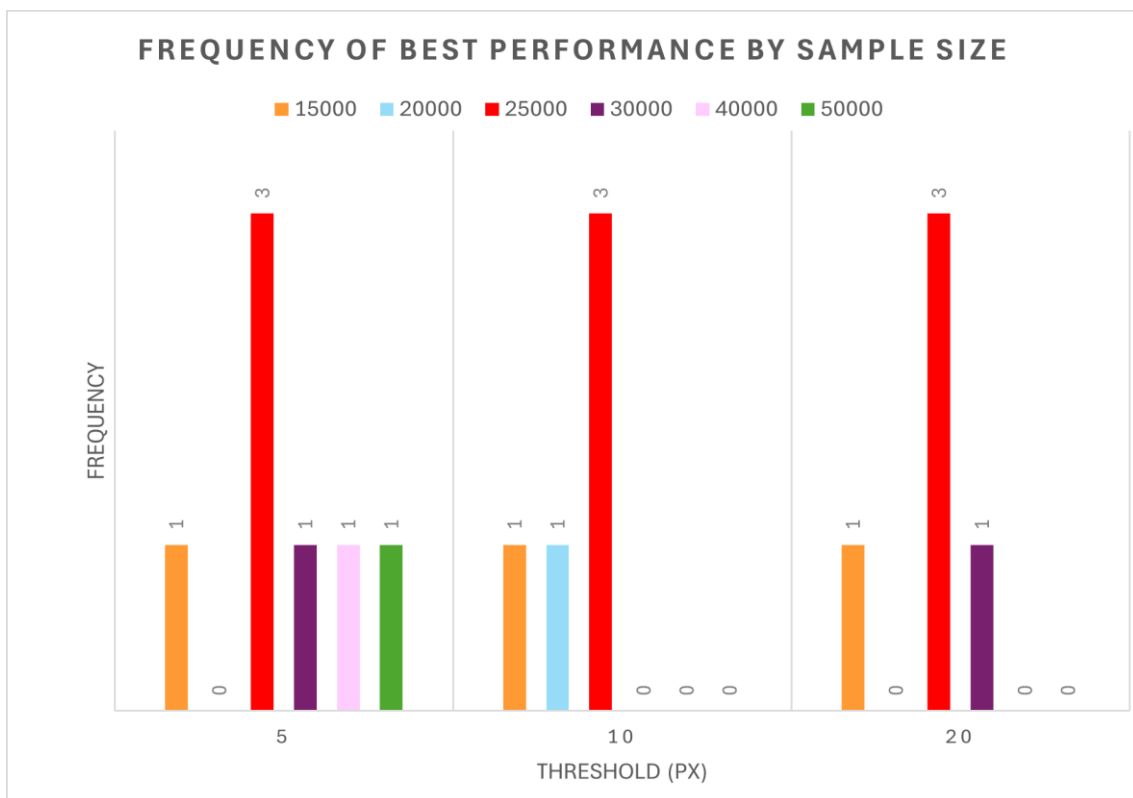


Figure 20 Frequency with which the sample sizes yield the best results across all methods. Multiple sample sizes can yield the best results for a certain threshold and method.

Figure 20 depicts how often a sample size achieves the best results. In total four comparative experiments were conducted, which marks the maximum number of best performances possible to achieve. A sample size of 25.000 achieves the best results three times across all thresholds. A sample size of 15.000 yields the best performance one time for each threshold. Other sample sizes only stand out one time for certain thresholds.

4 Discussion

The aim of this study was to evaluate the application potential of XFeat for cross-domain image matching. This was done by training XFeat on a RGB-MWIR dataset and comparing the performance of the two matching methods with respect to dataset rotation and sample size.

4.1 Impact of Sample Size on Matching Performance

Sample sizes of 5,000 and 10,000 always lead to inaccurate results. These sample sizes are too small to sufficiently train the network. The data does not properly represent the variety of the input imagery, respectively the network cannot learn all required information for identifying correct correspondences.

With larger sample sizes however this linear proportion of more samples leading to higher accuracies cannot be observed. The performance of the network trained with different amounts of sample sizes varies in inconsistent ways- more samples do not lead to more accurate results. While {15,000, 20,000, 25,000} samples outperform all greater sample sizes, {40,000, 45,000, 50,000} samples deliver better accuracy than {30,000, 35,000} samples (Figure 16). The greatest sample sizes of 45,000 and 50,000 never deliver the most accurate results.

For all methods and datasets medium sample sizes deliver the best matching results. Although the best performance oscillates between the above-mentioned sample sizes a peak can be identified at 25,000. This marks the point of overfitting. It confirms the non-linear relationship between data quantity and network performance for a permutation approach.

4.2 Impact of Dataset Rotation on Matching Performance

The experiments showed, that aligning the training data in the same direction delivers better results for the `match_xfeat` method but does not have an effect for the `match_xfeat_star` matcher. A reason for the significant improvement of accuracy for the `match_xfeat` method is that for this mutual nearest neighbour approach the identified keypoints should be each other's nearest neighbour to be correctly matched (Ying et al., 2025). This method is dependent on feature consistency, meaning that features should be similar regarding scale, rotation, perspective and lighting for instance. Aligning makes the cross-modal imagery a bit more similar, taking away the rotation inconsistency.

The Zero shot approach for lighter glue shows, that aligning the imagery significantly improves the matching results even if untrained on the characteristics of the dataset.

In practical applications this means that the flight direction must be at least coarsely known. Aligning relevant image data before feeding it into the network also increases computational effort.

4.3 Performance of the Matching Methods

In this study two matching methods were compared. A mutual nearest neighbour and a coarse to fine approach. The best performance was delivered by `match_xfeat` on a derotated dataset, with the RGB imagery aligned in flight direction. The MMA was four times better than the next best performance, delivered by `match_xfeat_star`. However, this method was also performing worst on a rotated dataset. For practical applications this means, that the imagery should be aligned if possible and be processed applying `match_xfeat_star`.

4.4 Limitations

Refeeding the same image pairs too often by expanding the sample size through permutation, increases the risk of overfitting. If the dataset cannot be expanded, the permutation strategy could be improved by adding variations to the imagery. Feasible approaches to expand training dataset size were presented by Wang et al. (2018) and Liu et al. (2018).

In this study 10,000 unique samples of matching RGB-MWIR image pairs were available. It is possible that certain patterns only show with higher sample sizes. For comparison: the study on H-Net by Liu et al. (2018) used a total of 160,000 pairs of cross-domain image patches as training data. Especially for more complex tasks like cross-modality matching a greater sample size is important.

The training data was all acquired within the same flight route. Hence, it does not reflect a great variation of landscape characteristics, besides seasonal changes. The imagery depicts very similar geometric structures. Overfitting might occur if a network is trained only on this dataset due to missing variation and sample size. The performance on different landscapes cannot be transferred from the results and must be assessed in another study.

To examine if XFeat can find correspondences and has potential to be further examined on the issue of cross-modality image matching the training data was sufficient.

The applied metrics in this study do not allow statements about the distinct positional accuracies of the correspondences. Also, the error threshold of 20 pixel is comparably high and must be carefully considered for future studies on the absolute accuracy of the matching results,

depending on the requirements of the application. A finer gradation of the error threshold would allow for more distinct results.

5 Conclusion

The experiments conducted in this study were a first guiding approach to access the potential of the neural network XFeat to match a cross-modality dataset of MWIR-RGB imagery. Another aim was to compare the performance of the matching methods to each other and evaluate the dataset rotation influence.

The results show, that the monomodal network XFeat can find correspondences even in a cross-modality dataset, the quantity of which depends on sample size, rotation and matching method. Correspondences are not random but learned.

The best performing method is using `match_xfeat` and aligning the training data in flight direction. Rotating the RGB reference map significantly increases the performance of `match_xfeat` but does not influence the performance of `match_xfeat_star`. On an unaligned dataset `match_xfeat_star` performs slightly better than `match_xfeat`.

Medium sample sizes of 15,000 to 25,000 with a peak for 25,000 perform best for all methods and thresholds.

5.1 Outlook

In this thesis the `lighterglue` matcher has been applied untrained on the dataset. Future studies should further investigate the performance of the `lighterglue` matcher by training it on the cross-modality dataset. Due to its attention mechanism, it has potential to outperform the purely convolutional network and matching methods presented in this thesis.

Further studies should focus on adjusting the hyperparameters of the network and assessing the impact on the matching performance.

Adapting the network architecture based on the concepts discussed in chapter 1.3 to enhance performance without compromising runtime and hardware efficiency is another promising approach.

Another interesting topic is the performance of the network on other training data, with datasets distinguishing between rural and urban landscapes or topography types.

6 References

- Aguilera, C., Aguilera, F., Sappa, A., Aguilera, C., & Toledo, R. (2016). *Learning Cross-Spectral Similarity Measures with Deep Convolutional Neural Networks*. <https://doi.org/10.1109/CVPRW.2016.40>
- Aguilera, C. A., Sappa, A. D., Aguilera, C., & Toledo, R. (2017). Cross-Spectral Local Descriptors via Quadruplet Network. *Sensors*, 17(4).
- Balntas, V., Johns, E., Tang, L., & Mikolajczyk, K. (2016). PN-Net: Conjoined Triple Deep Network for Learning Local Image Descriptors. <https://doi.org/10.48550/arXiv.1601.05030>
- Baruch, E., & Keller, Y. (2018). *Multimodal matching using a Hybrid Convolutional Neural Network*. <https://doi.org/10.48550/arXiv.1810.12941>
- Bello, I., Zoph, B., Le, Q., Vaswani, A., & Shlens, J. (2019, 27 Oct.-2 Nov. 2019). Attention Augmented Convolutional Networks. 2019 IEEE/CVF International Conference on Computer Vision (ICCV),
- Brown, M., Hua, G., & Winder, S. (2011). Discriminative Learning of Local Image Descriptors. *IEEE transactions on pattern analysis and machine intelligence*, 33(1), 43-57. <https://doi.org/10.1109/TPAMI.2010.54>
- de Rosa, G. H., & Papa, J. P. (2022). Chapter 7 - Learning to weight similarity measures with Siamese networks: a case study on optimum-path forest☆☆The authors appreciate São Paulo Research Foundation (FAPESP) grants #2013/07375-0, #2014/12236-1, #2017/25908-6, #2018/15597-6, #2018/21934-5 and #2019/02205-5, and CNPq grants 307066/2017-7 and 427968/2018-6. In A. X. Falcão & J. P. Papa (Eds.), *Optimum-Path Forest* (pp. 155-173). Academic Press. <https://doi.org/https://doi.org/10.1016/B978-0-12-822688-9.00015-3>
- Deng, H., Birdal, T., & Ilic, S. (2018). Ppfnet: Global context aware local features for robust 3d point matching. Proceedings of the IEEE conference on computer vision and pattern recognition,
- Deng, Y., & Ma, J. (2022). ReDFeat: Recoupling detection and description for multimodal feature learning. *IEEE Transactions on Image Processing*, 32, 591-602.
- DeTone, D., Malisiewicz, T., & Rabinovich, A. (2018, 18-22 June 2018). SuperPoint: Self-Supervised Interest Point Detection and Description. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW),
- Ding, L., Bawany, M. H., Kuriyan, A. E., Ramchandran, R. S., Wykoff, C. C., & Sharma, G. (2020). A Novel Deep Learning Pipeline for Retinal Vessel Detection In Fluorescein Angiography. *IEEE Transactions on Image Processing*, 29, 6561-6573. <https://doi.org/10.1109/TIP.2020.2991530>
- Dlesk, A., Vach, K., & Pavelka, K. (2021). Transformations in the Photogrammetric Co-Processing of Thermal Infrared Images and RGB Images. *Sensors*, 21, 5061. <https://doi.org/10.3390/s21155061>

- Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., & Sattler, T. (2019a). *D2-Net: A Trainable CNN for Joint Description and Detection of Local Features*. <https://doi.org/10.1109/CVPR.2019.00828>
- Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., & Sattler, T. (2019b). D2-net: A trainable cnn for joint detection and description of local features. *arXiv preprint arXiv:1905.03561*.
- Ebadi, A., Kaur, M., & Liu, Q. (2025). Hyperparameter optimization and neural architecture search algorithms for graph Neural Networks in cheminformatics. *Computational Materials Science*, 254, 113904. <https://doi.org/https://doi.org/10.1016/j.commatsci.2025.113904>
- Elharrouss, O., Mahmood, Y., Bechqito, Y., Serhani, M., Badidi, E., Riffi, J., & Tairi, H. (2025). Loss Functions in Deep Learning: A Comprehensive Review. <https://doi.org/10.48550/arXiv.2504.04242>
- En, S., Lechervy, A., & Jurie, F. (2018). *TS-NET: Combining Modality Specific and Common Features for Multimodal Patch Matching*. <https://doi.org/10.1109/ICIP.2018.8451804>
- Gerber, M., & Pillay, N. (2022). Automated Design of the Deep Neural Network Pipeline. *Applied Sciences*, 12(23).
- GitHub. (2024). Retrieved 12.12.2025 from https://github.com/verlab/accelerated_features/tree/main
- Hassaballah, M., Ali, A., & Alshazly, H. (2016). Image Features Detection, Description and Matching. In (Vol. 630, pp. 11-45). https://doi.org/10.1007/978-3-319-28854-3_2
- He, H., Chen, M., Chen, T., & Li, D. (2018). Matching of Remote Sensing Images with Complex Background Variations via Siamese Convolutional Neural Network. *Remote Sensing*, 10(2).
- Huang, Q., Guo, X., Wang, Y., Sun, H., & Yang, L. (2024). A survey of feature matching methods. *IET Image Processing*, 18(6), 1385-1410. <https://doi.org/https://doi.org/10.1049/ipr2.13032>
- Jin, Y., Mishkin, D., Mishchuk, A., Matas, J., Fua, P., Yi, K. M., & Trulls, E. (2021). Image Matching Across Wide Baselines: From Paper to Practice [Article]. *International Journal of Computer Vision*, 129(2), 517-547. <https://doi.org/10.1007/s11263-020-01385-0>
- Kaya, M., & BİLge, H. Ş. (2019). Deep Metric Learning: A Survey. *Symmetry*, 11(9).
- Lazebnik, S., Schmid, C., & Ponce, J. (2005). A sparse texture representation using local affine region. *IEEE transactions on pattern analysis and machine intelligence*, 27, 1265-1278. <https://doi.org/10.1109/TPAMI.2005.151>
- Li, J., Li, X., Wei, Y., Song, M., & Wang, X. (2022). Multi-Level Feature Aggregation-Based Joint Keypoint Detection and Description. *Computers, Materials & Continua*, 73, 2529-2540. <https://doi.org/10.32604/cmc.2022.029542>
- Li, L., Han, L., Gao, K., He, H., Wang, L., & Li, J. (2023). Coarse-to-fine matching via cross fusion of satellite images. *International Journal of Applied Earth Observation and Geoinformation*, 125, 103574. <https://doi.org/https://doi.org/10.1016/j.jag.2023.103574>

- Li, L., Liu, M., Ma, L., & Han, L. (2022). Cross-Modal feature description for remote sensing image matching. *International Journal of Applied Earth Observation and Geoinformation*, 112, 102964. <https://doi.org/https://doi.org/10.1016/j.jag.2022.102964>
- Li, X., Feng, Y., Xianguo, Y., Cong, Y., & Chen, L. (2025). Epipolar constraint-guided differentiable keypoint detection and description. *The Visual Computer*, 41, 7109-7121. <https://doi.org/10.1007/s00371-024-03795-4>
- Li, Z., & Snavely, N. (2018). Megadepth: Learning single-view depth prediction from internet photos. Proceedings of the IEEE conference on computer vision and pattern recognition,
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. European conference on computer vision,
- Lindenberger, P., Sarlin, P.-E., & Pollefeys, M. (2023). Lightglue: Local feature matching at light speed. Proceedings of the IEEE/CVF International Conference on Computer Vision,
- Liu, W., Shen, X., Wang, C., Zhang, Z., Wen, C., & Li, J. (2018). *H-Net: Neural Network for Cross-domain Image Patch Matching*. <https://doi.org/10.24963/ijcai.2018/119>
- Liu, Y., Sun, Z., Yu, B., Zhao, Y., Du, B., Xu, Y., & Cheng, J. (2025). MIFNet: Learning Modality-Invariant Features for Generalizable Multimodal Image Matching. *IEEE Transactions on Image Processing*, 34, 3593-3608. <https://doi.org/10.1109/TIP.2025.3574937>
- Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2), 91-110. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>
- Makosso, T. L., Almaktoof, A., & Abo-Al-Ez, K. (2025). Review of Different Types of Neural Network Architectures. *International Journal of Electrical Engineering and Applied Sciences (IJEEAS)*, 7(2). <https://doi.org/10.54554/ijeeas.2024.7.02.006>
- Mishchuk, A., Mishkin, D., Radenović, F., & Matas, J. (2017). Working hard to know your neighbor's margins: Local descriptor learning loss. <https://doi.org/10.48550/arXiv.1705.10872>
- Nasser, S. A., Gupte, N., & Sethi, A. (2024). Reverse knowledge distillation: Training a large model using a small one for retinal image matching on limited data. Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision,
- O'Shea, K., & Nash, R. (2015). An Introduction to Convolutional Neural Networks. *ArXiv e-prints*.
- Ono, Y., Trulls, E., Fua, P., & Yi, K. (2018). *LF-Net: Learning Local Features from Images*. <https://doi.org/10.48550/arXiv.1805.09662>
- Potje, G., Cadar, F., Araujo, A., Martins, R., & Nascimento, E. R. (2024, 16-22 June 2024). XFeat: Accelerated Features for Lightweight Image Matching. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),
- Revaud, J., De Souza, C., Humenberger, M., & Weinzaepfel, P. (2019). R2d2: Reliable and repeatable detector and descriptor. *Advances in neural information processing systems*, 32.

- Sarlin, P. E., DeTone, D., Malisiewicz, T., & Rabinovich, A. (2020, 13-19 June 2020). SuperGlue: Learning Feature Matching With Graph Neural Networks. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),
- Schmitt, M., Hughes, L. H., & Zhu, X. X. (2018). THE SEN1-2 DATASET FOR DEEP LEARNING IN SAR-OPTICAL DATA FUSION. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci., IV-1*, 141-146. <https://doi.org/10.5194/isprs-annals-IV-1-141-2018>
- Stolkin, R., Greig, A., & C, J. (2010). Measuring complete ground-truth data and error estimates for real video sequences, for performance evaluation of tracking, camera pose and motion estimation algorithms.
- Sun, J., Shen, Z., Wang, Y., Bao, H., & Zhou, X. (2021, 20-25 June 2021). LoFTR: Detector-Free Local Feature Matching with Transformers. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),
- Tian, Y., Fan, B., & Wu, F. (2017). *L2-Net: Deep Learning of Discriminative Patch Descriptor in Euclidean Space*. <https://doi.org/10.1109/CVPR.2017.649>
- Verykokou, S., & Ioannidis, C. (2025). Image Matching: A Comprehensive Overview of Conventional and Learning-Based Methods. *Encyclopedia*, 5(1).
- Wang, S., Quan, D., Liang, X., Ning, M., Guo, Y., & Jiao, L. (2018). A deep learning framework for remote sensing image registration. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145, 148-164. <https://doi.org/https://doi.org/10.1016/j.isprsjprs.2017.12.012>
- Xu, S., Chen, S., Xu, R., Wang, C., Lu, P., & Guo, L. (2024). Local feature matching using deep learning: A survey. *Information Fusion*, 107, 102344. <https://doi.org/10.1016/j.inffus.2024.102344>
- Yang, M., Wu, R., Yang, Y., Tao, L., Zhang, Y., Xie, Y., & Reddy, G. P. (2025). Image Matching: Foundations, State of the Art, and Future Directions. *Journal of Imaging*, 11(10), 329.
- Ye, Y., Shen, L., Hao, M., Wang, J., & Xu, Z. (2017). Robust Optical-to-SAR Image Matching Based on Shape Properties. *IEEE Geoscience and Remote Sensing Letters*, 14(4), 564-568. <https://doi.org/10.1109/LGRS.2017.2660067>
- Ying, S., Zhao, J., Li, G., & Dai, J. (2025). LIM: Lightweight Image Local Feature Matching. *Journal of imaging*, 11(5), 164. <https://doi.org/10.3390/jimaging11050164>
- Yu, C., Liu, Y., Zhao, J., Quan, D., & Shi, Z. (2024). Relational Representation Learning Network for Cross-Spectral Image Patch Matching. *ArXiv, abs/2403.11751*.
- Zagoruyko, S., & Komodakis, N. (2015). *Learning to compare image patches via convolutional neural networks*. <https://doi.org/10.1109/CVPR.2015.7299064>
- Zhang, H., Ni, W., Yan, W., Xiang, D., Wu, J., Yang, X., & Bian, H. (2019). Registration of Multimodal Remote Sensing Image Based on Deep Fully Convolutional Neural Network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, PP, 1-15. <https://doi.org/10.1109/JSTARS.2019.2916560>

Zhang, T., Hu, L., Li, L., & Navarro-Alarcon, D. (2021). Towards a multispectral RGB-IR-UV-D vision system—Seeing the invisible in 3D. 2021 IEEE International Conference on Robotics and Biomimetics (ROBIO),

Zhao, J., Yang, D., Li, Y., Xiao, P., & Yang, J. (2022). Intelligent Matching Method for Heterogeneous Remote Sensing Images Based on Style Transfer. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15, 6723-6731.
<https://doi.org/10.1109/JSTARS.2022.3197748>

Zhong, W., & Jiang, J. (2025). LGFCTR: Local and global feature convolutional transformer for image matching. *Expert Systems with Applications*, 270, 126393.

Zhu, J.-Y., Park, T., Isola, P., & Efros, A. (2017). *Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks*. <https://doi.org/10.1109/ICCV.2017.244>

Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., & Ling, H. (2021). Detection and tracking meet drones challenge. *IEEE transactions on pattern analysis and machine intelligence*, 44(11), 7380-7399.

Zhu, R., Yu, D., Ji, S., & Lu, M. (2019). Matching RGB and Infrared Remote Sensing Images with Densely-Connected Convolutional Neural Networks. *Remote Sensing*, 11(23).