Optimising Pixel-Wise Time-Series Analysis of Vegetation Indices Imagery via Key-Pixel Selection

Tibo Bruneel[1,2,*], Welf Löwe[1,2], Morgan Ericsson[2], Diego Perez-Palacin[2] and Jonas Nordqvist[3]

[1]*Softwerk AB, Reveljgränd 5 - Växjö, Sweden*
[2]*Dept. of Computer Science and Media Technology, Linnaeus University, Universitetsplatsen 1 - Växjö, Sweden*
[3]*Dept. of Mathematics, Linnaeus University, Universitetsplatsen 1 - Växjö, Sweden*

Abstract

Pixel-wise time-series analysis of satellite-derived vegetation indices is often complex and computationally demanding, as the pixel-wise application of the analysis scales with the number of pixels to be processed. In this study, we use Gaussian Process Regression (GPR) pixel-wise to fill temporal gaps in Normalized Difference Vegetation Index (NDVI) data. Rather than applying GPR across all pixels and trying to optimise the performance of GPR directly, we introduce an algorithm to select key pixels for GPR applications. By selectively applying GPR to a subset of pixels and image reconstruction of the remaining unselected pixels, we achieve substantial speedups ($2.5 - 5.5\times$) with minimal loss in NDVI estimation accuracy ($0.5 - 2.5\%$). This trade-off with pixel selection enables more scalable and efficient pixel-wise processing of large-scale imagery, with strong applicability in the field of remote sensing.

Keywords

Key-Pixel Selection, Pixel-Wise Processing, Performance Optimisation, Remote Sensing, NDVI, Temporal Gap-Filling, Gaussian Process Regression

1. Introduction

The process of pixel-wise time-series analysis involves processing a set of images distributed over time at the pixel level. Algorithms with higher complexities, especially when applied to fine-grained data, can run into problems meeting constraints on time and resources, as is common in production environments. The process becomes increasingly challenging as the size of the imagery scales, resulting in a vast number of pixels requiring individual processing. This can be particularly challenging when processing satellite imagery in remote sensing due to rapidly scaling data volumes across spatial and temporal dimensions, necessitating efficient processing and scalable analysis approaches.

We use NDVI imagery extracted from Sentinel-2 satellite data and Gaussian Process Regression (GPR) applied pixel-wise to fill gaps in time-series data caused by cloud cover, adverse weather conditions, or no satellite coverage. For each pixel changing its NDVI over time, a

SAIS2025: Swedish AI Society Workshop 2025, 16-17 June 2025, Halmstad, Sweden.
*Corresponding author.
tibo.bruneel@softwerk.se (T. Bruneel); welf.lowe@lnu.se (W. Löwe); morgan.ericsson@lnu.se (M. Ericsson); diego.perez@lnu.se (D. Perez-Palacin); jonas.nordqvist@lnu.se (J. Nordqvist)
0009-0000-4370-5981 (T. Bruneel); 0000-0002-7565-3714 (W. Löwe); 0000-0003-1173-5187 (M. Ericsson); 0000-0002-2736-845X (D. Perez-Palacin); 0000-0002-0510-6782 (J. Nordqvist)

corresponding GPR fills the gaps in between observations. It is cubic in the number of observations, which becomes problematic performance-wise when scaling up in data, *i.e.*, the number of pixels. In general, optimisations are often made on the algorithms, reducing their computational complexity, but often at the cost of accuracy, leading to a common trade-off between accuracy and computational performance. The same holds for GPR, where many significant optimisations have been introduced, but whilst losing accuracy in the process. Initial experiments revealed that such optimisations significantly impact accuracy on this gap-filling task. To address performance challenges, we explore adjusting the input data rather than modifying the algorithm, aiming to improve processing speed while preserving accuracy. Instead of applying GPR pixel-wise, we introduce a key-pixel selection algorithm and apply a three-step approach, introduced in Section 3. The motivation behind the three-step approach with pixel selection we present in this study is to improve the performance of GPR by using GPR less and having a more computational- and energy-efficient approach to pixel-wise processing.

2. Background

We briefly describe the data used in this study in Section 2.1 and provide a short introduction to Gaussian Process Regression in Section 2.2.

2.1. Data

In remote sensing, vegetation indices are used to assess the health, growth, and density of vegetation. They provide valuable insights for various applications, mainly agriculture and forestry monitoring. One of these indices is NDVI, a widely used indicator for monitoring vegetation health. NDVI ranges from -1 to 1, with $[-1, 0]$ indicating non-vegetated surfaces like water, snow, or barren land, up to $[0.5, 1]$ indicating dense, healthy vegetation like forests or crops. NDVI is calculated from satellite imagery by exploiting the contrast between red and near-infrared reflectance. Sentinel-2 provides high resolution imagery with 13 spectral bands, of which band 4 (RED) and band 8 (NIR) are used for deriving NDVI, with a precision of 10 m each, and a revisit rate of five days at the equator.

For the performance experiments in this study, we use data from three Scandinavian locations: Moelv (Norway), Lund (Sweden), and Salby (Denmark), with areas of 76, 103, and 138 hectares and pixel counts of 7826, 10478, and 14066, respectively. Three years of imagery were used (January 1st, 2021, until December 31st, 2023). We obtained raw Sentinel-2-L2A surface reflectance data [1], from which we then calculated NDVI data [2].

2.2. Gaussian Process Regression

Gaussian Process Regression (GPR) [3, 4, 5] is a Bayesian, non-parametric approach to regression that models distributions over functions through a mean function and a kernel, which controls the function's smoothness and flexibility. Parameters are optimised to fit the data, often using gradient-based methods. GPR provides predictions along with uncertainty estimates by evaluating the posterior standard deviation. In this study, we use GPR to fill gaps in time-series data with large, irregularly spaced gaps.

3. Key-Pixel Selection Optimisation Approach

The overall optimisation strategy is to (i) select a subset of key pixels for which gap filling is performed, (ii) use an accurate GPR to fill the gaps, and (iii) use image reconstruction through spatial linear interpolation to fill the gaps for the pixels not selected in (i).

In section 3.1, we present the key pixel selection algorithm applied in (i), in section 3.2, the selection of its hyperparameters, and in section 3.3, the image reconstruction used in (iii). As the focus is on optimising the general performance, we do not delve into GPR details of (ii).

3.1. Key-Pixel Selection Algorithm

This algorithm selects pixels to process in (ii) by classifying each pixel into one of four categories: border, local deviation, filler, or non-processing. The classification determines whether the pixel should be gap-filled with (ii) GPR or with (iii) interpolation. All classifications but the non-processing pixels enter GPR processing. In post-processing, all the unprocessed pixels receive spatially interpolated values from surrounding processed pixels at each point in time.

- **Border Pixel Detection**: Identify and classify border pixels by checking if any neighbouring pixels only have unknown values across time. Due to irregularly shaped fields and incomplete satellite imagery, matrix edges do not reliably define borders, so actual borders must be detected. The detection itself may be omitted when a mask is available.
- **Local Deviation Pixel Detection**: For non-border pixels, identify and classify pixels with neighbours showing an average difference over time exceeding a threshold hyperparameter. This classification targets high variance regions or inter-field borders to retain local information that could otherwise get lost through spatial interpolation, such as roads, and water streams. The condition for a pixel (x, y) can be formulated as following:

$$\max_{(x',y')\in N(x,y)} \frac{1}{T}\sum_{t=1}^{T} \left|I(x', y', t) - I(x, y, t)\right| > \theta,$$

  where $I(x, y, t)$ represents a pixel value at coordinate x, y and timestamp t, $N(x, y)$ represents the set of neighbours, T represents the number of timestamps, and θ represents the deviation threshold parameter.
- **Filler Pixel Detection**: Identify and classify the remaining pixels as filler pixels based on a predefined distance from the previously classified pixels, where the distance is a hyperparameter. The filler pixel is assigned in regions that do not have borders or local deviation pixels assigned to the nearest neighbors. This stage is targeted at filling regions with too few selected pixels, leading to too large spatial interpolations.
- **Non-processing Assignment**: Assign any remaining unclassified pixels to the non-processing category. These pixels are omitted for GPR and later receive values from the spatial interpolation.

Once the classification algorithm is completed, GPR is applied on all the border, local deviation, and filler pixels. In figure 1, the classifications are visualised, (a) highlights the border pixels, (b) highlights the local deviation pixels, (c) highlights the filler pixels, and (d) highlights the three classifications together, representing all pixels to be processed by GPR.

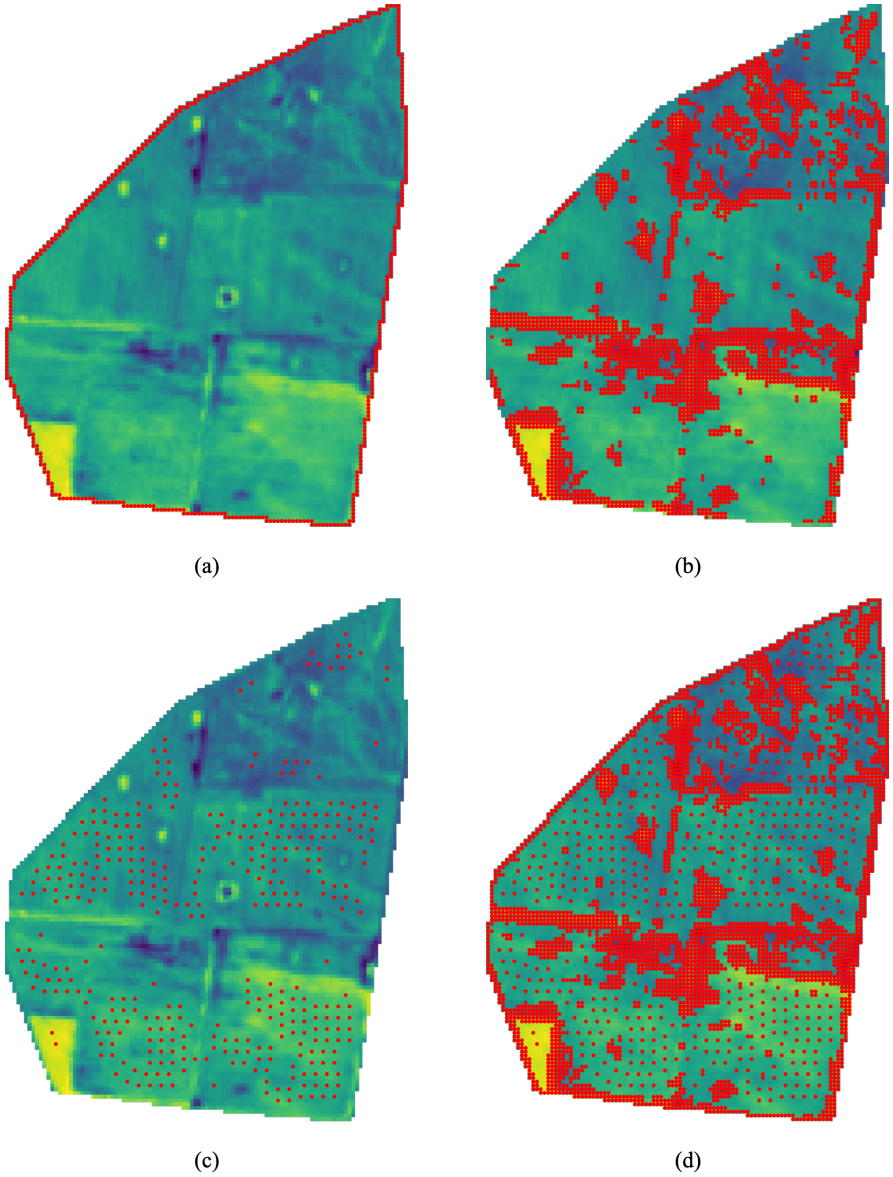
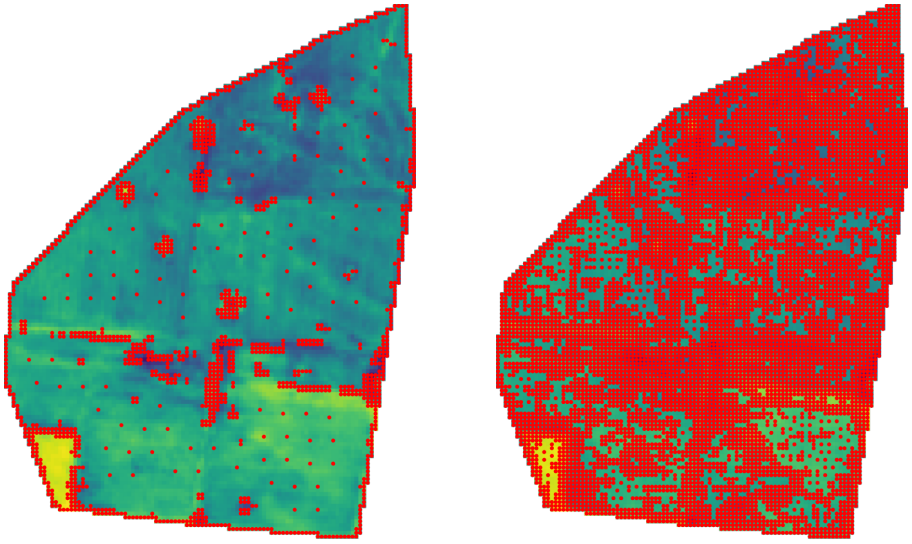


Figure 1: (a) NDVI image with border pixels highlighted. (b) NDVI image with local deviation pixels highlighted. (c) NDVI image with filler pixels highlighted. (d) NDVI image with all GPR selected pixels.

3.2. Hyperparameters and Selection

The algorithm uses two hyperparameters: *deviation threshold*, which determines the deviation in (NDVI) values between it and its neighbours for the local deviation pixel detection, and *filler distance*, defining the pixel distance for assigning filler pixel classifications. Lower deviation thresholds and filler distances increase the number of selected pixels and the processing needs, while higher values may reduce processing time at the cost of accuracy. For optimal tuning, one should consider the data scale, desired detail, and variance over time, especially since off-season

data typically has a lower variance. When using both on- and off-season data, the algorithm can be adjusted to only look at the largest differences or only at on-season data if such information is available. Figure 2 illustrates pixel selection under more extreme hyperparameter settings. On (a), excessively strict parameters result in too few pixels being selected, potentially causing substantial information loss in post-processing. On (b), overly relaxed and sensitive parameters select an excessive number of pixels, minimising information loss but significantly increasing computational demands, thus largely diminishing the algorithm's potential efficiency gains.



(a) (b)

Figure 2: (a) NDVI image with highly strict parameters resulting in too few pixels selected. (b) NDVI image with overly sensitive parameters, resulting in an excessive number of pixels selected.

Parametrising NDVI data is relatively straightforward due to its normalised value range, constrained between $[-1, 1]$. This bounded range enables swift tuning of parameters, such as via grid search, across representative datasets, often yielding generalisable results with minimal experimentation.

When setting the local deviation parameter, it is important to account for data noise. Sentinel-2 NDVI data typically exhibits noise below 0.05 in value. Therefore, setting the deviation parameter to at least 0.05 puts it safely above the noise floor, capturing genuine deviations within the imagery. Similarly, the spatial correlation between pixels must be considered for the filler pixel distance parameter. Physical NDVI correlation between two points diminishes over distance, and it is generally unreasonable to assume that pixels more than 100 meters apart (distance of 10 pixels) are strongly correlated or correlated at all. A filler pixel distance in the range of 2–5 pixels (20–50 meters) is therefore typically more appropriate.

The tuning of both parameters can further depend strongly on physical variables, for example, more robust parameters may be required in highly variable data, whereas more stable parameters may suffice in more uniform and less variable data.

3.3. Image Reconstruction through Spatial Interpolation

After applying GPR to the selected pixels and completing all heavy gap-filling computation, selected pixels now have values across the entire time series. However, unselected pixels remain without any values under the temporal gaps since they were excluded from the GPR analysis. This issue can be solved through image reconstruction.

Image reconstruction refers to the process of estimating missing or corrupt pixel values in imagery using available information from surrounding pixels. At each time step, unselected pixels are reconstructed based on the values of the nearby processed pixels. This work evaluates only linear spatial interpolation experimentally due to its simplicity and effectiveness for the task. Specifically, reconstruction is performed by linearly interpolating between the values of neighbouring pixels with known GPR-estimated values. As a result, all previously missing pixels under temporal gaps receive interpolated estimates derived from the GPR-filled data.

4. Experimental Setup

We conduct performance experiments on the three data locations by applying the Gaussian Process Regression on pixels selected by the algorithm. We provide an overview of the key-pixel selection and GPR setup in Section 4.1, and describe the evaluation of the approach in Section 4.2.

4.1. Key-Pixel Selection and GPR Setup

To evaluate the key-pixel selection algorithm, we explore a range of hyperparameter configurations. A full grid search is conducted over combinations of deviation threshold values {0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4} and filler pixel distances {2, 3, 5, 10}. These settings are not chosen for direct hyperparameter optimisation, but rather to assess the effect of hyperparameter variations on NDVI error and computational speed-up, therefore some parameters might not be appropriate for a real world setting.

The GPR setup remains consistent across all iterations and pixel selection hyperparameters. Known NDVI data points serve as training data, where x represents time values and y denotes NDVI values. We employ a GP model with a Radial Basis Function (RBF) kernel, initially set with a length scale of 32, which is further scaled and optimised during training. Model parameters are learned through gradient descent using the Adam optimiser based on the Marginal Log Likelihood. As previously highlighted, the task with GPR in this study is to interpolate temporal gaps, where the learning aim is to minimise the difference between the GPR estimations and the available ground truth NDVI values.

4.2. Evaluation

The proposed three-step approach is evaluated based on two key performance metrics: estimation error and computational speedup.

First, the estimation error is assessed by comparing final estimated values, derived from GPR or interpolation, against ground truth NDVI values. This quantifies the loss in accuracy introduced by the three-step approach relative to pixel-wise processing. The Mean Absolute

Error (MAE) is used here, calculated as the average of all absolute errors between available ground truth points and estimations at the corresponding points in time.

Second, computational speedup measures the factors of reduction in computing time. It indicates how many times faster the optimised process runs compared to standard pixel-wise processing. This factor is computed by timing the full execution of both the three-step approach and the baseline pixel-wise method, then calculating how many times faster the optimised approach runs compared to the baseline.

Initially, we perform the experiment pixel-wise to establish baseline MAE and computing time. Subsequently, the pixel selection is applied under each parameter setting, with both metrics recorded for each. Each experiment is repeated 10 times across all settings and datasets, the mean values across all iterations are used in evaluation to minimise potential bias.

5. Experiment Results & Discussion

In this section we present the experiment results in Section 5.1, and provide a more general discussion on results in Section 5.2.

5.1. Results

The evaluation results for all three locations are visualised in Figure 3, with the plots on the left showing the MAE based on the hyperparameters, while the plots on the right showcase the computational speedup compared to pixel-wise GPR based on the hyperparameters.

The evaluation results are similar at all three locations. As either the deviation threshold or filler pixel distance increases, the MAE generally increases, by less than 0.01 and up to 0.05, *i.e.*, between 0.5% to 2.5% of the NDVI value range. This is an expected effect, as fewer pixels are selected, leading to less GPR being applied and more pixels being interpolated. This, however, also results in a larger computational speed-up by 2.5 to 5.5 times. On all the locations, one can observe that with the low hyper-parameter configurations, only a slight MAE error of $\approx$ 1% is introduced whilst gaining a speed-up of $\approx 2.5\times$. After the biggest increase in error and speedup, the metrics flatten out, and the hyperparameters no longer have as much effect as they did in lower parameter values, with speedups from 3 to 5.5 factors and absolute errors up to 0.09 NDVI in the worst hyperparameter configurations, an increase of about 0.05 NDVI error compared to pixel-wise baseline, which is more than double an increase in NDVI error.

5.2. Discussion

The results presented in Figure 3 reveal a clear trade-off between estimations accuracy and computational efficiency across all three locations. Despite this trade-off, the results suggest that with a balanced configuration, a substantial improvement in speedup is achieved for a minimal loss in accuracy, as can be seen with the lower parameter configurations on the locations. Such a balanced configuration is particularly promising for operational deployment in a production setting, where efficiency is critical, and a slight error can be tolerated. The effect of this is large, as *e.g.*, on Denmark, pixel-wise GPR takes $\approx$ 250 seconds on the used engine, but using pixel

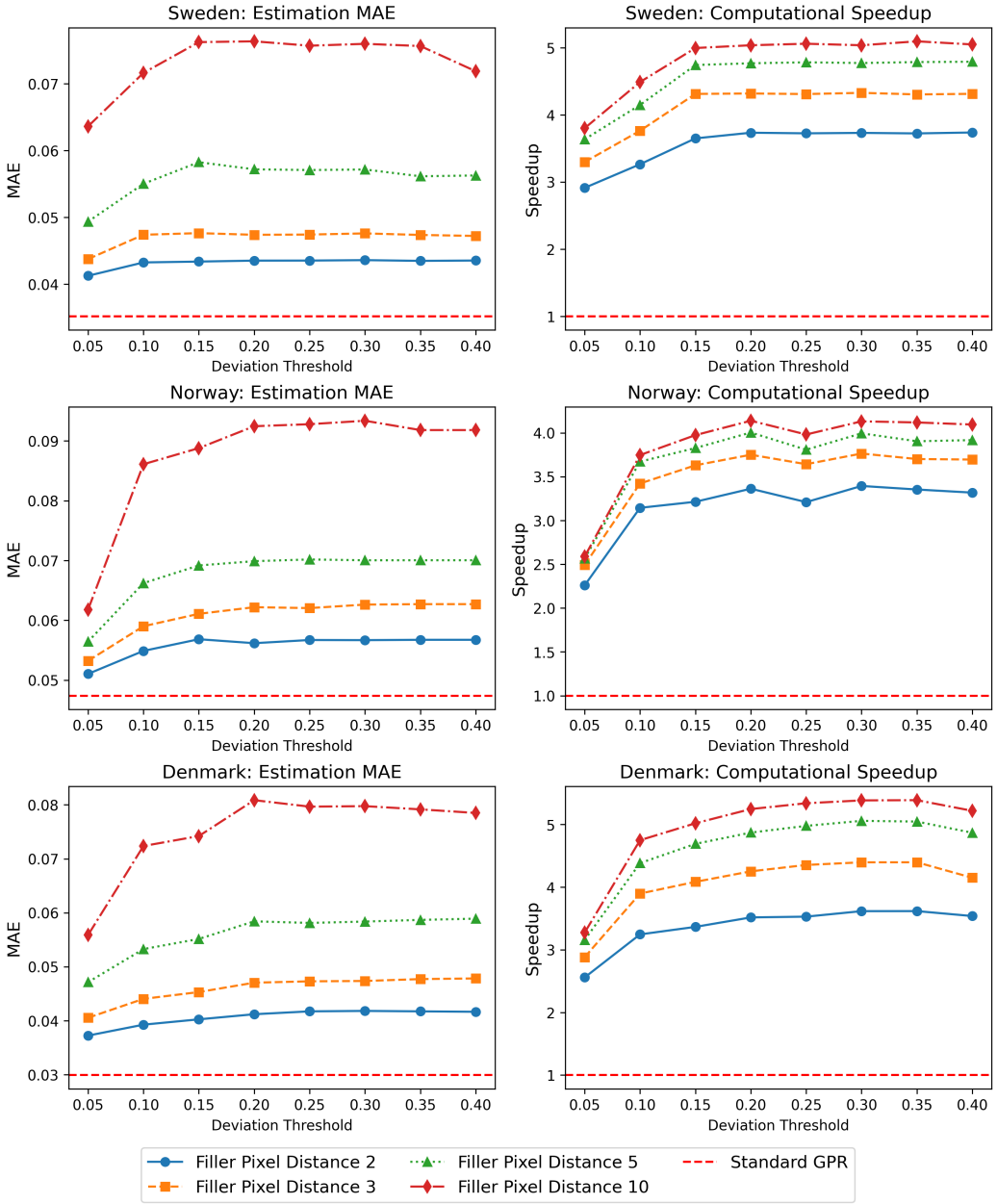


Figure 3: Mean Absolute Error (MAE) and computational speedup of applying GPR on the complete imagery using pixel selection across varying deviation thresholds and filler pixel distances for Sweden, Norway, and Denmark.

selection with the lowest parameter configurations achieves a 2.5× speed-up, reducing it to ≈ 100 seconds, with only a 0.008 increase in NDVI MAE.

The results also highlight that controlling this balance is rather flexible with the parameter settings, such that it can be fine-tuned with ease depending on the requirements *i.e.*, accuracy

and resources. The approach shows consistent behaviour across Sweden, Norway, and Denmark, suggesting that the method generalises well over the Northern Europe region. Since the method relies on statistical properties, we expect it to also be applicable towards other regions, vegetation types, and other vegetation indices, with the necessary parameter tuning. However, empirical validation for generalisability remains an important direction for future work.

Despite its strengths, the approach is not without limitations. The inherent trade-off between accuracy and efficiency requires careful tuning; poorly chosen parameters may result in excessive information loss, *e.g.*, a filler pixel distance of 10 while offering high speed-up, leads to a notable degradation of several factors in accuracy that may not be acceptable in many applications. Hence, optimal configurations are typically found in the parameter space before the performance metrics begin to plateau, where there is a trade-off between accuracy and speedup that is worth it based on the application.

6. Related Work

This section provides an overview of related work, beginning with pixel selection methods in Section 6.1, connecting to the key-pixel selection algorithm presented in this paper (Section 3.1). This is followed by a short discussion on image reconstruction in Section 6.2, connecting to the post-processing stage of our three-step approach (Section 3.3).

6.1. Pixel Selection

Pixel selection in a spatio-temporal setting involves identifying highly informative or relevant pixels from a stack of time-series images. What qualifies as "informative" depends on the specific task; for example, in change or anomaly detection, the focus is on identifying pixels that exhibit sudden shifts or unusual variations over time.

In this paper, the application is data reduction, where the aim is to preserve the major patterns in the imagery while minimising data redundancy. In this context, an informative pixel is one that contributes relatively more to the overall variance, structure, or dynamics of the data than others. For spatio-temporal NDVI datasets, this information spans both spatial and temporal dimensions. Therefore, key-pixel selection seeks to retain critical spatial structures and temporal dynamics while discarding redundant data.

The remainder of this subsection provides a brief overview of some existing spatio-temporal data reduction techniques, dividing it in sampling-based techniques in Section 6.1.1, on which the key-pixel selection in this paper is based, other non-sampling techniques in Section 6.1.2 that could possibly be used as part of the optimisation approach, and a comparison of these techniques with the presented key-pixel selection in this work in Section 6.1.3.

6.1.1. Sampling-Based Techniques

Sampling-based techniques reduce data volume by sampling a representative subset of pixels. Common approaches include:

- **Random sampling**: Selects pixels with equal probability.

- **Stratified sampling**: Divides data into strata—defined spatially, temporally, or both, and samples from each stratum.
- **Importance sampling**: Favours regions of higher interest based on data characteristics.
- **Deterministic sampling**: Selects pixels based on conditions. An approach that yields consistent results for the same input.
- **Grid sampling**: Partitions the space, time, or both dimensions into regular grid cells and selects a fixed representative (*e.g.*, the centre point) subset of pixels from each.

Naïve techniques, particularly pure random selection, often overlook underlying patterns and dynamics, limiting their effectiveness in real-world applications [6]. Rule- and hyperparameter-based approaches are more sensitive to these dynamics, but may require fine-tuning to capture representative subsets of pixels effectively.

6.1.2. Non-Sampling Techniques

Clustering-based techniques group pixels with similar spatio-temporal data characteristics while ensuring sufficient dissimilarity between clusters. Superpixel-based techniques group neighbouring pixels into spatially coherent regions called superpixels, which preserve important structural information. Unlike clustering techniques, superpixel techniques emphasize spatial contiguity, ensuring that the pixels selected in these regions are spatially connected; clustering techniques do not necessarily preserve this. Once a cluster or superpixel has been formed, a representative subset of pixels can be selected from it, *e.g.*, centroids. Various clustering algorithms like K-means and DBSCAN [7, 8] can be adapted for working with the spatio-temporal data and the task of pixel selection and data reduction [9, 10, 8]. Superpixels are commonly used in image segmentation tasks but can also be adapted for pixel selection [11, 12].

Deep learning offers a powerful and flexible framework that can be used for identifying informative pixels in spatio-temporal imagery, as neural networks[1] can be leveraged to learn complex patterns and extract meaningful features from the data [18, 19].

6.1.3. Comparison to Presented Key-Pixel Selection Algorithm

The introduced key-pixel selection algorithm in this work combines both deterministic sampling used in border and local deviation pixel detection, and grid sampling used in filler pixel detection. The aim of the deterministic sampling step is to select pixels of high spatio-temporal importance, where the grid sampling is used to cover areas where the deterministic criteria may undersample. Together, these sampling strategies complement each other, as deterministic sampling alone may leave larger areas completely unselected, but grid sampling alone may omit many highly informative pixels.

Compared to clustering, superpixel-based techniques, or deep learning methods, the proposed approach is significantly more lightweight, more interpretable, and easier to tune. It operates

[1] Convolutional Neural Networks (CNNs) [13] are effective for capturing spatial features and structures in imagery. Recurrent Neural Networks (RNNs) [14] are well suited for handling sequences and capturing temporal dynamics. Along with variants like LSTM [15] and GRU [16], addressing limitations of standard RNNs, particularly in learning long-range dependencies. Transformer architectures [17], having been most prominent in language processing, could be used for the capturing of global context and long-range dependencies of the imagery series.

with only two hyperparameters and applies deterministic rules based on raw NDVI values rather than derived or normalised features, preserving a direct link between the selection logic and the physical characteristics of the data. This enhances the explainability a lot. Additionally, the approach has very low complexity, requires minimal memory or computational resources, and is easily parallelisable for larger spatial regions, as it has no dependencies beyond close neighbouring pixels.

The method has limitations compared to alternative methods. Its selection quality depends heavily on set parameters and it has limited adaptivity to input data as it assumes a more uniform data structure (*e.g.* single environmental type). On top of that, the method does not incorporate higher-level semantic understanding or parametrised memory, features inherently available in deep learning.

6.2. Image Reconstruction

Image reconstruction can be performed on a single image, where the reconstruction relies solely on the spatial neighbourhood of a pixel, with the assumption that nearby pixels tend to exhibit similar characteristics or provide enough information to estimate the missing pixels. In such cases, simpler methods can be used, such as linear interpolation in this paper. Other common interpolation methods include cubic interpolation, which uses non-linear estimates based on surrounding known pixel values, and nearest-neighbour interpolation, which assigns each unknown pixel the value of its closest known neighbour.

Image reconstruction can also be extended beyond spatial context to include temporal context with hybrid approaches or other datasets with fusion approaches, making the reconstruction further multi-dimensional [20].

7. Conclusion & Future Work

7.1. Conclusion

There is no free lunch when optimising performance on more complex methods. However, sometimes one can get a highly discounted lunch. In this study, we presented a simple approach to improving the performance of pixel-wise analysis on NDVI imagery. Instead of applying GPR pixel-wise, we select key pixels to perform GPR on and then spatially interpolate the unselected pixels in post-processing. By doing so, we observe a speedup of at least $\approx 2.5\times$ as a trade-off for slightly more error in processed images. This initial study on the three-step approach with pixel selection shows promising results on performance optimisation when being constrained to pixel-wise processing. The approach becomes especially attractive with analysis methods that have higher complexities, such as GPR. The impact of such a rather simple approach can be very large and scales with the usage of pixel-wise analysis, further reducing costs, minimising time to analysis, and lowering computational & energy requirements by several factors.

While this work is grounded in the field of remote sensing, it contributes and is generalisable to the field of artificial intelligence by demonstrating how input-level optimisation, through selective pixel processing, can reduce computational demands without minimal compromise to

machine learning model performance. This strategy aligns with broader efforts around efficient inference, resource-aware learning, and scalable deployment of machine learning models.

7.2. Future Work

To assess the generalisation capability of the three-step approach with pixel selection, future research could evaluate its performance in regions beyond Northern Europe, covering a broader range of vegetation types and land cover across diverse environments. Future work can also explore other datasets than NDVI and other processing methods than GPR, or rather investigate the application of GPR further, including variants like sparse and variational GPR, and higher-dimensional GPR. The key-pixel selection approach could possibly be extended with adaptive parametrisation based on data characteristics to mitigate poor manual parametrisation, or be extended with more advanced image reconstruction methods in the future as well.

Acknowledgments

Tibo Bruneel's work was funded by the Industry Graduate School on "Data Intensive Applications (DIA)" at Linnaeus University, which is partially funded by the Knowledge Foundation. Additional funding for the research and this study was provided by Softwerk AB and Vultus AB. Sentinel-2 data used in this work was made available through Vultus AB.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] Sentinel Hub, Sentinel hub: Cloud api for satellite data, 2024. URL: https://www.sentinel-hub.com/, accessed: 2024-11-01.

[2] C. J. Tucker, Red and photographic infrared linear combinations for monitoring vegetation, Remote Sensing of Environment 8 (1979) 127–150. URL: https://www.sciencedirect.com/science/article/pii/0034425779900130. doi:https://doi.org/10.1016/0034-4257(79)90013-0.

[3] D. G. Krige, A statistical approach to some mine valuation and allied problems on the witwatersrand, 1951. URL: http://hdl.handle.net/10539/17975, thesis (M.Sc.(Engineering)).

[4] C. E. Rasmussen, C. K. I. Williams, Gaussian processes for machine learning, 2008.

[5] J. Wang, An intuitive tutorial to Gaussian process regression, Computing in Science & Engineering 25 (2023) 4–11.

[6] M. Whelan, N. A. L. Khac, M.-T. Kechadi, Data reduction in very large spatio-temporal datasets, in: 2010 19th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises, 2010, pp. 104–109. doi:10.1109/WETICE.2010.23.

[7] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: Proceedings of the Second International

Conference on Knowledge Discovery and Data Mining (KDD-96), AAAI Press, 1996, pp. 226–231.

[8] M. Y. Ansari, A. Ahmad, S. S. Khan, G. Bhushan, Mainuddin, Spatiotemporal clustering: a review, Artificial Intelligence Review 53 (2020) 2381–2423. URL: https://doi.org/10.1007/s10462-019-09736-1. doi:10.1007/s10462-019-09736-1.

[9] J. Han, M. Kamber, J. Pei, Data Mining: Concepts and Techniques, 3rd ed., Morgan Kaufmann, 2011.

[10] Z. Shi, L. S. Pun-Cheng, Spatiotemporal data clustering: A survey of methods, ISPRS International Journal of Geo-Information 8 (2019). URL: https://www.mdpi.com/2220-9964/8/3/112. doi:10.3390/ijgi8030112.

[11] I. B. Barcelos, F. D. C. Belém, L. D. M. João, Z. K. G. D. Patrocínio, A. X. Falcão, S. J. F. Guimarães, A comprehensive review and new taxonomy on superpixel segmentation, ACM Comput. Surv. 56 (2024). URL: https://doi.org/10.1145/3652509. doi:10.1145/3652509.

[12] S. Subudhi, R. N. Patro, P. K. Biswal, F. Dell'Acqua, A survey on superpixel segmentation as a preprocessing step in hyperspectral image analysis, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14 (2021) 5015–5035. doi:10.1109/JSTARS.2021.3076005.

[13] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (1998) 2278–2324. doi:10.1109/5.726791.

[14] J. L. Elman, Finding structure in time, Cognitive science 14 (1990) 179–211.

[15] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (1997) 1735–1780. URL: https://doi.org/10.1162/neco.1997.9.8.1735. doi:10.1162/neco.1997.9.8.1735.

[16] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using rnn encoder-decoder for statistical machine translation, 2014. URL: https://arxiv.org/abs/1406.1078. arXiv:1406.1078.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[18] J. Li, Y. Cai, Q. Li, M. Kou, T. Z. and, A review of remote sensing image segmentation by deep learning methods, International Journal of Digital Earth 17 (2024) 2328827. URL: https://doi.org/10.1080/17538947.2024.2328827. doi:10.1080/17538947.2024.2328827. arXiv:https://doi.org/10.1080/17538947.2024.2328827.

[19] L. Wang, M. Zhang, X. Gao, W. Shi, Advances and challenges in deep learning-based change detection for remote sensing images: A review through various learning paradigms, Remote Sensing 16 (2024). URL: https://www.mdpi.com/2072-4292/16/5/804. doi:10.3390/rs16050804.

[20] Q. Wang, Y. Tang, Y. Ge, H. Xie, X. Tong, P. M. Atkinson, A comprehensive review of spatial-temporal-spectral information reconstruction techniques, Science of Remote Sensing 8 (2023) 100102. URL: https://www.sciencedirect.com/science/article/pii/S2666017223000275. doi:https://doi.org/10.1016/j.srs.2023.100102.