



AALBORG
UNIVERSITY

Machine Learning for Automatic Detection of Historic Stone Walls Using LiDAR Data

Ana Cristina Mosebo Fernandes & Ezra Francis Leslie Trotter
Aalborg University | Masters in Geoinformatics | 4 June 2021

NIRAS

Preface

This thesis is submitted as part of the final assessment in the Masters in Geoinformatics program at Aalborg University, Denmark. The work in this thesis was completed between the 1st February and the 4th June 2021.

This thesis seeks to assess the feasibility of, and limitations in, applying Convolutional Neural Networks and LiDAR elevation data to the identification of culturally significant stone walls in the Danish landscape.

We the authors would like to thank our supervisor at Aalborg University Carsten Keßler. We would also like to thank our colleagues at the Geodata Department at NIRAS, in particular Casper Fibæk whose generous support allowed the project to come to fruition.

Individually, we would each like to thank our partners for their continuous support and for providing constant motivation.

Abbreviations

CLC	Corine Land Cover
CNN	Convolutional Neural Network
DEM	Digital Elevation Model
DHM	<i>Danmarks Højdemodel</i>
DL	Deep Learning
DSM	Digital Surface Model
DTM	Digital Terrain Model
EPSG	European Petroleum Survey Group
ETRS	European Terrestrial Reference System
FCCN	Fully Convolutional Neural Network
FCN	Fully Convolutional layers
HAT	Height Above Terrain
LIDAR	Light Detection and Ranging
LULC	Land Use and Land Cover
MAE	Mean Absolute Error
MSE	Mean Squared Error
NDVI	Normalized Difference Vegetation Index
R-CNN	Residual - Convolutional Neural Network
ReLU	Rectified Linear Unit
RMSE	Root Mean Squared Error
SDFE	<i>Styrelsen for Dataforsyning og Effektivisering</i>
UTM	Universal Transverse Mercator

List of Figures

Fig. 1 - Different shapes and forms of stone walls and sand walls, in its original form (a and b) and how they look like today (c and d).....	8
Fig. 2 - Study site: the municipality of Ærø.	12
Fig. 3 - Stone walls and DTM in Ærø	13
Fig. 4 - Stone walls registered per year in Ærø	13
Fig. 5 - Land Cover types where stone walls are located, in Denmark (top) and Ærø (down).	14
Fig. 6 - DTM in Hillshade with stone walls dataset (red)	14
Fig. 7 - Examples of stone wall profiles. The top three profiles show the presence of stone walls, while the bottom three are examples of profiles where a wall is no longer present	15
Fig. 8 - Classification model architecture used	17
Fig. 9 - Example of a stone wall anti-aliased	18
Fig. 10 – Example the location and size of 64x64 pixel patches in the label data	18
Fig. 11 - Model architecture of the CNN model with a shallower structure compared to the U-Net original model	19
Fig. 12 - Loss and validation loss with calculated median for all the final model iterations	22
Fig. 13 - Example of patches of stone wall predictions and respective labels and ground truth	23
Fig. 14 - Examples of model predictions	24
Fig. 15 - Difference between prediction with absence data (left) and without (right)	24
Fig. 16 - Difference of predictions between using DTM alone (left), DTM and HAT (center), and the final model (DTM, Sobel filter and HAT) (left).	25
Fig. 17 - Example of External validation predictions in Silkeborg	25
Fig. 18 - Stone walls predicted as removed or damaged after postprocessing.....	26
Fig. 19 - Examples of post-processed found walls in Ærø.....	27
Fig. 20 – Onsite Photo 1.	27
Fig. 21 – On-site Photo 2.....	27
Fig. 22 - Example of a prototype for a WebGIS to visualize the results of the predictions	28
Fig. 23 - On site photos for location 1.....	36
Fig. 24 - On site photos for location 2.....	37
Fig. 25 - On site photos for location 3.....	37
Fig. 26 - On site photos for location 4.....	38
Fig. 27 - On site photos for location 5.....	38
Fig. 28 - On site photos for location 6.....	39

List of Tables

Table 1 - Summary statistics of the model for training, validation and evaluation (test) data.....	22
Table 2 - Evaluation metric results for different training datasets.....	22

Contents

Abbreviations.....	3
List of Figures.....	4
List of Tables.....	4
Abstract.....	6
1. Introduction.....	7
2. Related Work.....	9
2.1 LiDAR Data.....	9
2.2 Deep Learning.....	9
3. Materials.....	11
3.1 Terrain Data.....	11
3.2 Study Site and Stone wall Dataset.....	12
4. Method.....	14
4.1 Pre-process of Stone wall Dataset.....	14
4.2 Profile Based Machine Learning.....	16
4.3 Data Preparation and image patches generation.....	17
4.4 Model Training and Prediction.....	19
4.5 Assessment and Post-Processing.....	21
5. Results.....	22
5.1 Quantitative Assessment.....	22
5.2 Qualitative Assessment and External Validation.....	23
5.3 Postprocessing.....	26
6. Discussion.....	28
7. Conclusion.....	30
8. References.....	32
9. Annex 1 – Field site validation.....	36

Abstract

Stone walls are structures present in the landscape of Denmark and are protected not only for their cultural and historical significance, but also for their vital role in supporting local biodiversity. A considerable number of stone walls structures have either disappeared, suffered substantial damage, or had segments removed. Additionally, as it stands today, the registry of these structures, managed by each municipality respectively, is not up-to-date and lacks completeness. Recent developments in Machine Learning and CNN's, and the increasing availability of LiDAR-based (Light Detection and Ranging) terrain models have enabled new methodologies in the extraction and mapping of terrain features and structures. With this in mind, our study aims to analyse the publicly available terrain data derived from the Danish LiDAR data (40 cm resolution), using a U-Net-like CNN model, in order to assess the stone walls dataset, and provide for an update of the registry. The study was focused on the Danish municipality of Ærø. Good results were seen using the Digital Terrain Model (DTM) alone, however better results were obtained when adding Height Above Terrain (HAT) and an additional DTM layer with a Sobel filter applied. Using a pixel-wise evaluation, there was an overall agreement of 93% between ground truth and prediction of stone walls in a validation area, and 88% overall agreement for the whole predicted area. Good generalizability was found when externally validating the model on new data, showing positive results for either the existent stone walls, as well as predicting new potential ones, upon visualization. The method performed best in open areas, however positive results were also seen in forested areas, although denser areas and urban areas presented as challenging. Given the inexistence of a reference dataset or other studies on this specific matter, the evaluation of our study was heavily based on the stone walls registry itself, and visual inspection of the predictions and on the ground. Further improvements can come from the inclusion of aerial imagery and other relevant data, as well as further optimization of the CNN model. This application demonstrates the potential of automatization the process of identification and update of the stone walls' registry in Denmark, of great relevance to the local governments. We suggest that a Decision Support System be developed to allow municipalities access to the results of this method.

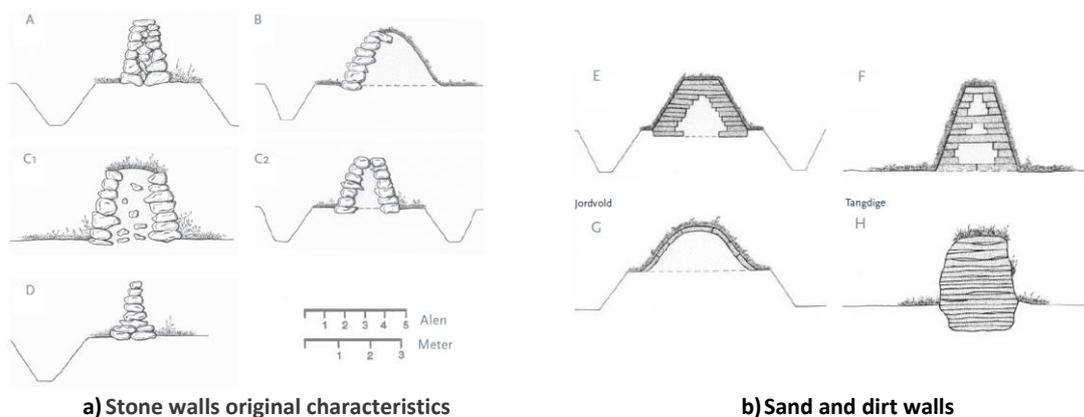
Keywords: stone walls; LiDAR; DEM; Convolutional Neural Networks; UNet; Deep Learning; structure detection; topographic analysis

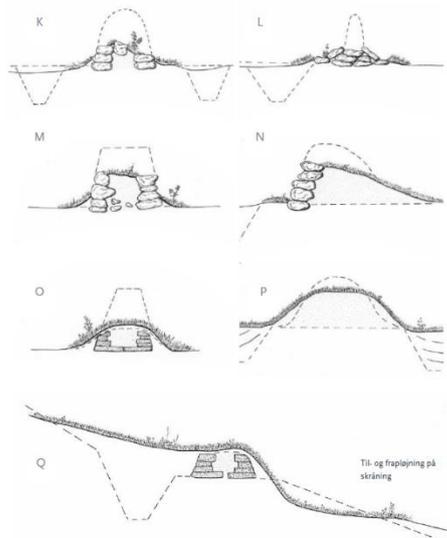
1. Introduction

Stone walls are structures that populate the landscape of Denmark and are protected for their cultural and historical significance. The most recent structures were built in the 1800s and were used to divide and mark agriculture fields during agrarian reforms. Additionally, they were used to mark forests and woods which had been planted by regal decree at the time [1]. The oldest structures were built in the first century, where they most commonly were erected as property boundaries and to mark administrative divisions. In addition to their intended purpose, stone walls also serve as vital green corridors between landscapes [2]. Therefore, these structures are important to preserve not only for their historical importance, but also for their vital role in supporting local biodiversity.

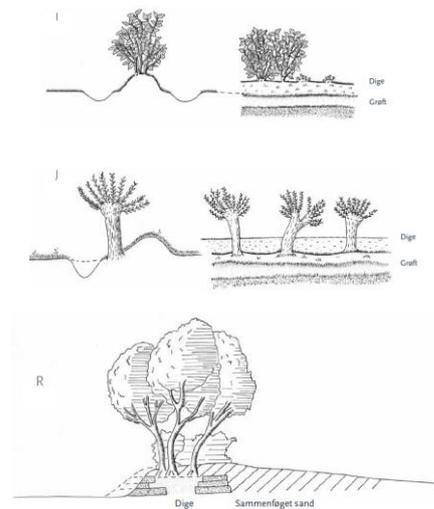
As a consequence of urban expansion and development of agricultural land, a substantial number of Denmark's stone wall structures have either already disappeared or have suffered considerable damage [1]. As a result of this, Danish stone walls and sand walls were classified as protected in 1992 by the nature protection law [3], which was followed by an update in 2004 with the "Museums' law", which transferred responsibility to the individual municipalities. Today, each municipality is responsible for the management and protection of its culturally significant stone walls and sand walls. Additionally, it is the municipalities responsibility to inform the Ministry of Culture of the stone walls' locations so they can be updated in the ministry's central registry. However, as it stands today, the Danish Ministry of Culture dataset is not necessarily up to date and, in all likelihood does not fully account for all the stone walls and sand walls in each municipality [4]. Recent acknowledgement of this, triggered reaction from the Ministry to set up a "reference group" to research methods to update the registry in an automated fashion.

The definition of stone walls by the law characterizes them as "Man-made, linear elevations of stone, earth, turf, seaweed or similar materials which function or have functioned as fences and have or have had the purpose of marking administrative property or use boundaries in the landscape" [1]. Protected stone walls include the structures falling under this definition, as well as those already registered in the 1:25 000 topographic map [5], in public domain and those that are situated on or near protected habitats. Their physical characteristics vary in size, shape and materials. Generally, the walls are between 0.5 and 1.5m in height, 1.5m in width; made of either of stone, heather peat, earth or a combination thereof (Fig. 1).





c) Stone walls as they look today



d) Stone walls and dirt walls with different vegetation

Fig. 1 - Different shapes and forms of stone walls and sand walls, in its original form (a and b) and how they look like today (c and d). Source: [1]

According to official guidelines [1], structures located in dense urban areas don't fall under the protection classification by the Museums law, because of continuous urban expansion which has led to the removal of most of these structures. Besides urban areas, stone walls located on summerhouse areas, around churches, or protection dikes are generally not classified as protected.

The removal or partial destruction of protected stone walls is against the established law [3], however it is possible to apply for a dispensation to the respective municipality, who will assess the case for removal or alteration. For this, an up-to-date registry is necessary, to not only verify the structure in question, but also to monitor and ensure the preservation of the protected stone walls.

With this in mind, this study aims to analyse terrain data derived from the Danish LiDAR (Light Detection and Ranging) dataset, in order to assess the stone walls dataset, and provide for an update of the registry. For updating the registry, the study will focus on two main tasks:

- 1) Analyse stone walls with the terrain data, and identify stone walls or segments of walls that no longer are existent on the ground, and were removed;
- 2) Find and map potential new stone wall structures that are not registered but should be included in the registry.

To accomplish this, we will engage in the analysis of LiDAR derived data in order to profile the stone walls in Ærø, then we will use a simplified morphometric algorithm to identify their topographic peak points. This analysis will also validate the stone walls dataset and serve as feature-engineering to prepare the dataset for the second task. We will apply a Deep Learning method by using a Convolution Neural Network (CNN) model in order to find and map potential stone walls that are not registered, using the validated stone walls dataset. With this we intend to provide for an automated method for updating the registry of protected stone walls, which coupled with a visualization tool, as well as on-site confirmation by experts, could be of great use to the municipalities in Denmark. The rest of the study is structured as followed: Section 2 references previous work related to the data and methods used in similar contexts to this study; Section 3 describes the materials and methods utilized for accomplishing this work, namely the validation of the stone walls dataset and its pre-processing, the U-Net-like model for the prediction of new stone walls, as well as its post-processing; Section 4

provides the results obtained and their assessment; Section 5 engages in the discussion of the results and its applications; and finally Section 6 summarizes the work performed in this study.

2. Related Work

2.1 LiDAR Data

Stone walls are prominent features of the Danish landscape. They are visible in aerial imagery in clear, open landscapes; however, it is harder to observe them when they are located in woods and forests. Therefore, elevation models and derived LiDAR data provide a valuable means to identify structures and objects on the surface, which collect data by measuring their range and reflectance and are not affected in the same way by vegetation [6]. In the field of archaeology, many studies focus on the identification of structures amongst terrain. A myriad of Archaeology mapping studies use derived elevation models from LiDAR data, to identify objects and structures on the topographic landscape [7] [8] [9]. This is because the ability of LiDAR data to penetrate forest and scrub canopies allows for the identification of structures on the surface that otherwise would remain hidden. Additionally, the high spatial resolution has provided new insights on more detailed landscapes, providing knowledge and context to archaeological studies. The increase of the availability of LiDAR data has also contributed to the expansion of the applicability of such data to many fields of research. This availability is also mirrored by the implementation of more ambitious projects, such as the LiDAR scanning of the entirety of the Earth surface, intended by the Earth Archive project [10], to create an open-source digital twin of the planet, to document the land surface at a high resolution.

Furthermore, other studies use LiDAR data to identify and control vegetation distribution [11], the mapping of height vegetation [12], and research on biodiversity and support nature management [13]. These studies explore the impact and importance of topography for different aspects of biodiversity, by analysing different LiDAR-based measures such as terrain slope, roughness, aspect, hillshade or elevation above sea level, and their relationship with vegetation and local environment; emphasising their relevance and applicability. LiDAR-derived data have also been adopted for use in flood mapping and monitoring. The high resolution and accuracy of Digital Elevation models derived from LiDAR data are extremely important to identify critical areas at risk of flooding, due to their capability for identifying detailed features of the surface, hence with great impact on flood risk and hazard prediction [14].

The use of LiDAR derived data is therefore of great importance for planning and urban and land management. Its range of applicability is also justified by the time and cost effectiveness, which combined with automatization methods, allows for an optimization of otherwise time-consuming, expensive and labour-intensive tasks. LiDAR derived data facilitate the detection and extraction of land surface objects, especially at a finer scale, can allow for example the identification of land cover patterns [15] [16].

2.2 Deep Learning

Neural Networks (NNs) have been used in remote sensing and image analysis increasingly in recent years in a variety of applications. Deep Learning (DL) algorithms are an extension of NNs and have been used for image classification, object detection, segmentation, change detection, amongst other tasks [17]. Its successful performance during such tasks has helped in the automation of processes in many applications, and has proven to be a useful and valuable method of analysis and automation.

Convolutional Neural Networks (CNN) are a sub-class of DL methods, and are used in state-of-the-art computer vision tasks. A CNN is a learning model which takes an image as input data and assigns importance to various objects in the image [18]. A CNN is usually composed of convolution layers,

pooling layers and fully connected layers, performing feature extraction and generating a new feature map for data prediction as final output, either as classification or regression. During training of a CNN, units that are organized in a feature map are connected to local patches of units from the previous layer, by a set of weights (filters), and each of them go through a non-linear transformation, a process repeated for each convolutional layer. These set of weights, or filters, are called convolutional kernels, that go through the image as a moving window. Each convolution layer has many convolutional kernels with variable weights, extracting different feature maps. The performance of the kernels and weights is tracked by a loss function, calculated by forward propagation on the training data. These parameters are optimized during training, according to the loss value, to minimize the difference between the output of the model and the ground truth data, performed by backpropagation and gradient descent as optimization algorithms. Both convolution and pooling layers perform feature extraction, although the role of pooling layer is to extract and merge features with similar characteristics into one, by either computing the maximum (or the average), reducing the size of the representation (down-sampling), as well as the number of parameters to compute, helping in this way to control overfitting [18]. Fully Connected layers map the features extracted by the previous convolution and pooling layers to the final outputs of the network, followed by an activation function, which is selected depending on the requirements of the specific task.

CNNs have been widely used to solve classification problems, such as Land Use and Land Cover (LULC) classification, and analysing hyperspectral data and high spatial resolution images, with a high rate of classification accuracy [19]. For LULC studies such as [19], a deep neural network structure is used where image patches are the input into the network and features are extracted using convolutions layers followed by normalization, an activation layer and pooling, connecting to a fully connected layer that will then classify into the respective land cover classes. According to [20] [21], image patch-based CNN is more suitable than pixel-based CNN, when using medium-resolution images, given the lack of fine detail.

Other methods of classification, namely semantic segmentation and instance segmentation, use a pixel-wise classification, and are also widely used in remote sensing and image analysis. These methods use a fully convolution network, with fully convolution layers (FCN), where the network layers are up-sampled after down-sampling, usually producing an output of equal spatial dimensions to the input data [22]. These layers are also designated as encoding and decoding layers. An example of FCNs is U-Net, originally developed for semantic segmentation of medical images, first introduced in 2015 [23]. [24] has used semantic segmentation to detect surface disturbance caused by mining from topographic maps, where the study mentions a high level of accuracy achieved. The study uses a modified U-Net architecture, where the model first uses an encoder component (contracting path) whose feature maps learn spatial patterns at different scales, by utilizing blocks of 2-D convolutional layers followed by max pooling operations, obtaining a low-dimensional representation of the input image. These feature maps are then used in the decoder (also known as deconvolution, or expansive path) layers, to convert back to the original dimension of the image, to produce the final output segmentation map. In the mentioned study, a classification is performed at each pixel location, where the output is the probability of each pixel belonging to each of the classes defined in the model. This method differs from instance segmentation, where the goal is not only to identify the object from the background, but to identify and attribute a label to each pixel, as well as an individual object of a label. [25], proposes instance segmentation to map topographic features using LiDAR-derived data (slopes). The study explores the application of Mask R-CNN and mentions successful accuracies with such method. Mask R-CNN is a derivation of R-CNN, which applied convolution to regions of interest in the image, instead of its entirety. The Mask R-CNN extension allows for polygon masks to be generated in each region of interest, outputting the predicted class for the region, the final

predicted bounding box and the predicted segmentation. The results of the study suggest that the method performs well when applied to LiDAR data, focusing on features with distinctive topographic and geomorphic characteristics.

The methods applied to the research problem in focus is much dependent on data availability, the representation of the data and the problem itself. Defining a clear initial problem statement will dictate what would be the desired output for the DL model, thereby narrowing down the selection of the model architecture and problem according to the task at hand. However, data availability and its representation greatly affect the performance of the DL algorithm [17]. With a small dataset, the model might not be able to cope with bias (underfitting) or variance (overfitting), and lose its ability to fit the data and generalize well.

Data quality is essential in achieving good model performance and significant results. While dealing with datasets that represent real world applications, it is all too common to have noisy and faulty data, often necessitating rigorous pre-processing. Neglecting the importance of data processing and preparation can lead to data cascades [26], where data issues cause downstream effects, leading to poor model performance and output. In our study, the stone walls dataset is a representation of the stone wall structures present in the Danish landscape, whose quality is not fit-for-purpose. The dataset is neither an accurate representation the stone wall's precise geographic location, nor is it current with the ground truth. Such dataset characteristics determine the design and structure of the project and the approach to the problem statement. In order to update the stone walls registry, we will first engage in verifying the presence of the individual walls against the Danish elevation model, since most of the structures are easily observable in the terrain data. Here we intend to remove sections of stone wall that no longer exist, and adjust the position of the walls in the dataset in order to better reflect their actual location on the ground. We will then use the validated data to detect and map potential non-registered stone walls, by performing a regression task using a Deep Learning model with a U-Net-like architecture.

3. Materials

3.1 Terrain Data

Terrain data for Denmark is made publicly available by The Danish Map supply¹, which is the distribution channel for the Agency of Data supply and Efficiency (SDFE). A digital terrain model (DHM/*Terræn* 2014) [27] and a digital surface model (DHM/*Overflade* 2014) [28] were used in this study, both of which were downloaded from The Danish Map supply's website, and are based on aerial laser scanning measurements taken in December of 2014. Both datasets describe Denmark's surface in relation to mean sea-level and are provided as raster layers with a pixel size of 0.4m. For this study, the DSM refers to the representation of bare earth and the objects on the surface (trees, vegetation, buildings, etc.), while DTM is the representation of bare earth surface. Both datasets cover the entirety of Denmark and are available as a 32-bit GeoTiff, with a pixel resolution of 0.4m. Each individual pixel value is precise to 0.15m in the horizontal direction and 0.05m in the vertical direction, and were made available in the ETRS89 UTM 32N coordinate system (EPSG: 25832) [29]. All the data in this study had the same local projection.

Additionally, a third dataset was created by subtracting the pixel-wise value of the DSM from that of the DTM to give the Height Above Terrain (HAT), also known as normalized DSM (nDSM) [6]. This was done over the extent of each raster, thereby creating a new raster of the same size, with pixel values 'HAT = DSM – DTM'. The HAT can describe the height of structures above the surface, providing height

¹ Accessible at <https://download.kortforsyningen.dk/>

information on individual objects, such as buildings, and in this case on stone wall structures. The analysis of the HAT alone was not sufficient to identify stone walls, as they proved hard to distinguish from other linear structures. However, given its potential of providing additional context of the stone wall structures' location, it was considered as an additional layer to the training data.

Additionally, a Sobel filter was used in conjunction with the DTM to create a new layer. The Sobel operation implements a 2D spatial gradient calculation on an image by sliding a pair of convolution masks (3x3) on the *x*-direction (horizontal) and on the *y*-direction (vertical), respectively [30]. The mask manipulates the pixels one by one, changing the value of the pixel according to the kernels. The Sobel layer was created in Python using the Buteo toolbox, accessible through the project repository <https://github.com/casperfibaek/buteo>.

3.2 Study Site and Stone wall Dataset

To reduce the amount of data required and overall processing time, a smaller study area was selected. The island Municipality of Ærø lies in the Baltic Sea between the Danish Island of Funen and the German Region of Schleswig and has an area of 88km² (Fig. 2). It was chosen for its smaller size relative to other municipalities, its abundance of stone walls, and its gently undulating landscape.

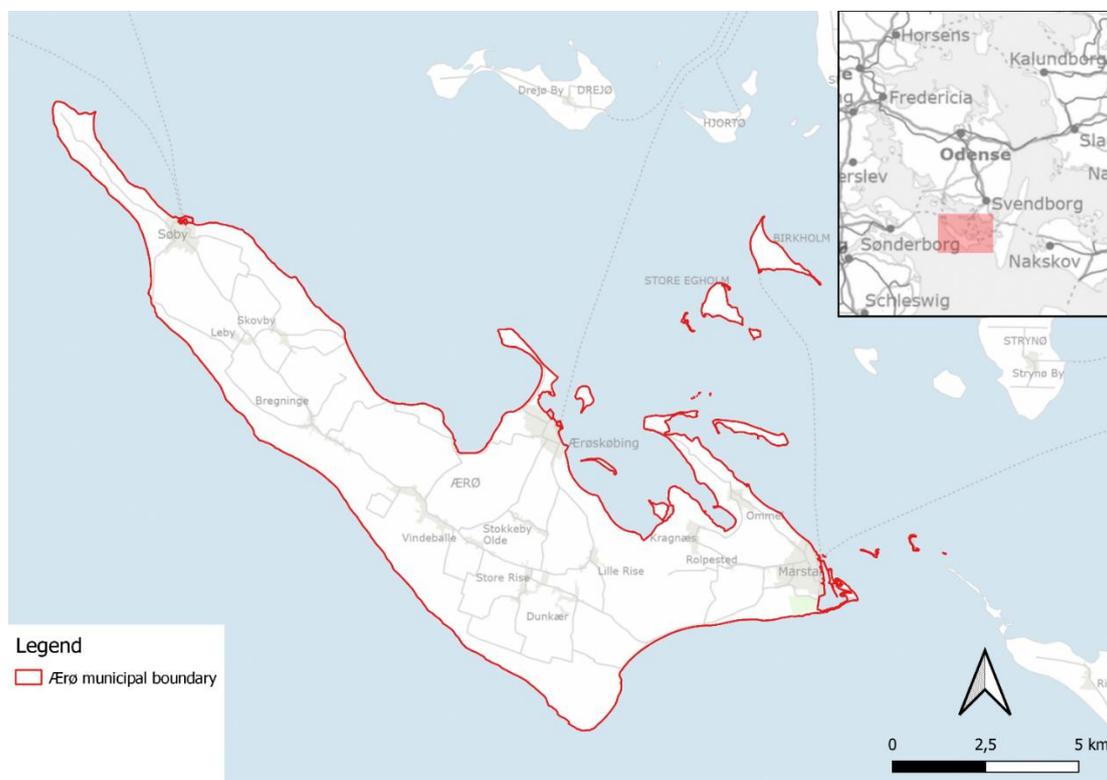


Fig. 2 - Study site: the municipality of Ærø.

A digitised Map of Denmark's protected stone walls is made available as part of the danish Ministry of Culture's stone wall registry. This map was first digitised on the 1st of July 1992 and updated in 2006, and is made publicly available as a vector dataset for download through the Ministry's data portal² [31]. On Ærø, the dataset contains 2766 stone walls for a total length of around 514 kilometres. Each wall in the dataset is represented as a vector linestring along with the walls' associated metadata. The metadata contains information such as the date of registry, the walls' current condition and the institution under responsibility. Given that the majority of stone walls on Ærø were registered as a

² Accessible at <https://arealdata.miljoportal.dk/>

result of the digitisation of the 1:25 000 topographic map of Denmark, most of the metadata is either incomplete or no longer current. This is especially critical in some areas where almost 30% of the protected stone walls did not appear in the subsequent versions of the topographic map [32]. However, there have been recent efforts to update the registry (Fig. 4).

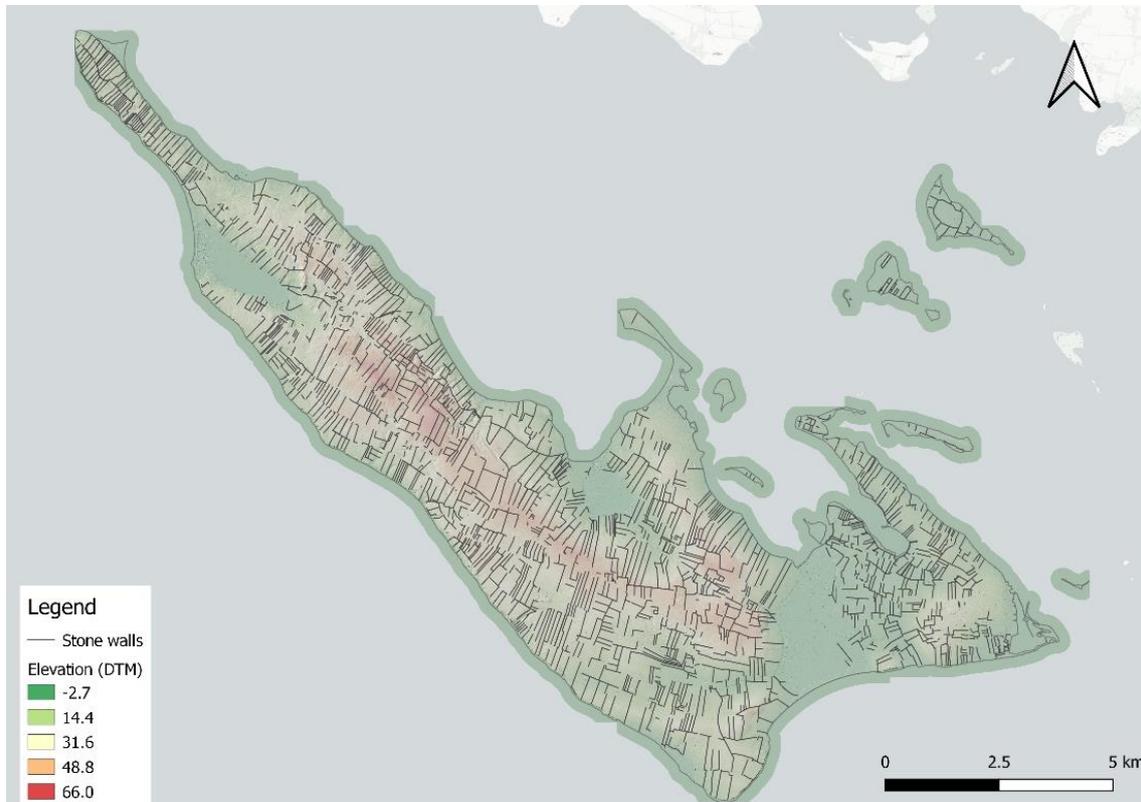


Fig. 3 - Stone walls and DTM in Ærø

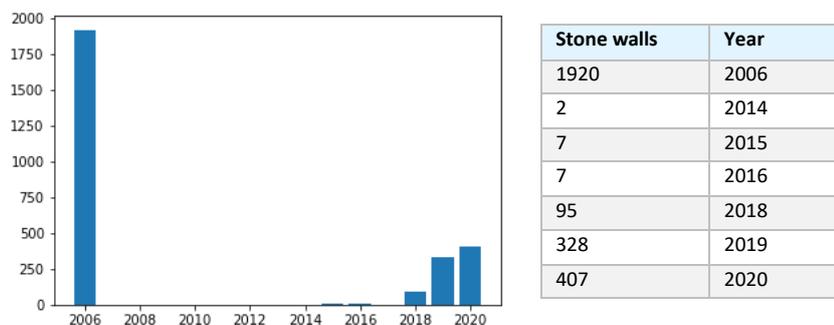


Fig. 4 - Stone walls registered per year in Ærø

An analysis of the 2012 Corine Land Cover (CLC) [33] sourced from SDFE’s data portal, reveals that stone walls on Ærø are found in generally similar types of landcover as those in the rest of Denmark. Through this analysis it was found that ~80% of the stone walls in the data set were located on agricultural land, both on Ærø and in the rest of Denmark. This similarity continues with ~5% of stone walls being located in discontinuous urban fabric, both on Ærø and Denmark (Fig. 5). Where Ærø differs is in the number of stone walls found in forested areas. The municipality has very little forested land, and as a result almost no stone walls are found in forests, whereas ~15% of Denmark’s stone walls are found in forested areas. It should be noted that in this analysis, each stone wall was only given one landcover category, however individual stone walls could traverse multiple landcover types.

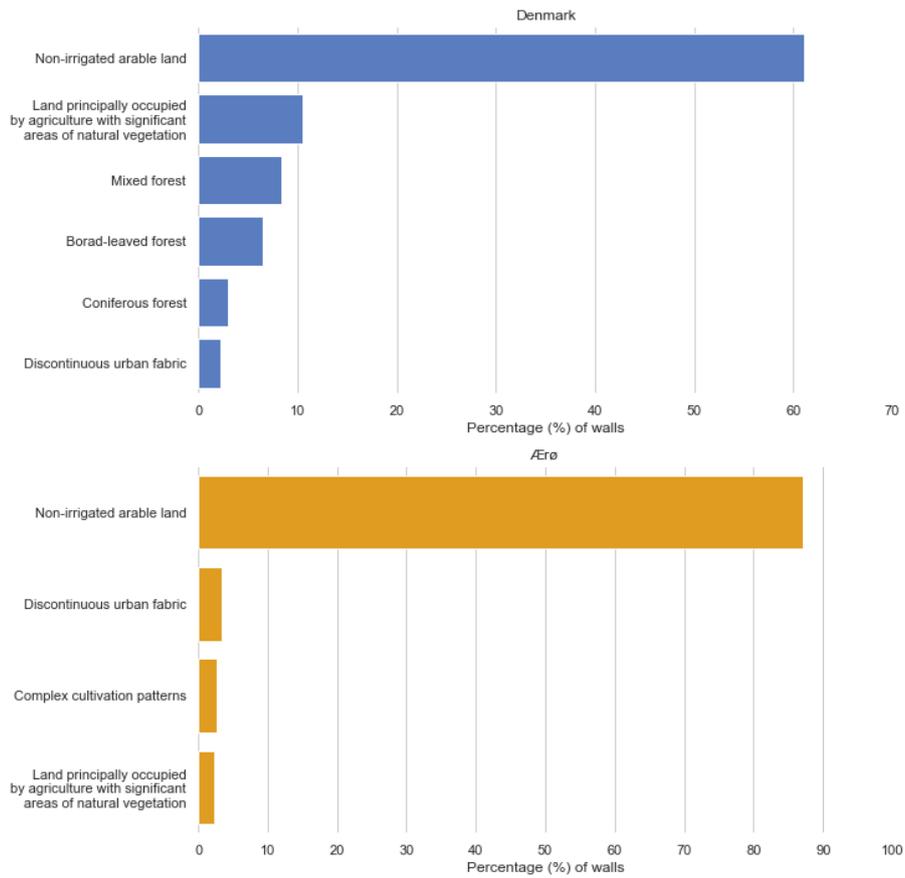


Fig. 5 - Land Cover types where stone walls are located, in Denmark (top) and Ærø (down).

4. Method

4.1 Pre-process of Stone wall Dataset

The initial step was to validate the stone wall dataset against the most recent DTM data. The Digital terrain Model is quite useful for detecting stone walls because of their topographic characteristics, standing out from their surrounding landscape. When performing a hillshade analysis, the structures are clearly visible (Fig. 6).

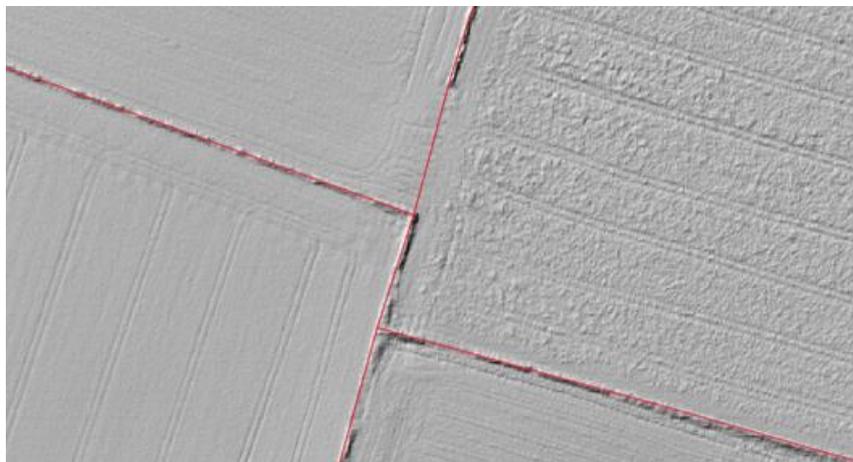


Fig. 6 - DTM in Hillshade with stone walls dataset (red)

This initial step is deemed as necessary given the difference between the two datasets regarding their production date. The DTM data was current as of 2014, while the stone wall reference dataset was digitised in 2006. As discussed earlier, when the reference was digitized there was little effort to validate the individual walls. Reference data representation accuracy has a large effect on the overall performance and results of a DL model [17], and it was important to find and remove stone walls segments that had been removed or altered in the intervening time, to achieve the best possible results during the later stages of our study.

The initial inspection and validation had four distinct steps:

- Step 1: Create profiles along each line representing each stone wall
- Step 2: Check each profile for the presence of stone wall
- Step 3: Redraw the dataset with absent walls removed
- Step 3: Validation of the corrected dataset

Step 1: Creating Profiles

The initial step taken was to segment each line in the dataset into 5m sections, with the purpose of creating a profile at the end of each section. Some walls were represented as straight linestrings and others were multi-linestring objects. The presence of multi-linestring objects required an initial step of segmenting each multi-linestring into its composite lines. This was necessitated by the later process of recreating the dataset in the adjusted positions. At each 5m subsection, a tangential line is created (cross-section) with a length of 10m, which is then broken up into 0.4m subsections, which corresponds to the pixel resolution of the DTM. At each of these points, the elevation value of the DTM is extracted, and a 3D multipoint object was created containing the x and y positional coordinates of the points in each profile, along with the value for the elevation. The initial dataset of 2.766 lines yielded 113.089 profiles, each comprising of 50 points.

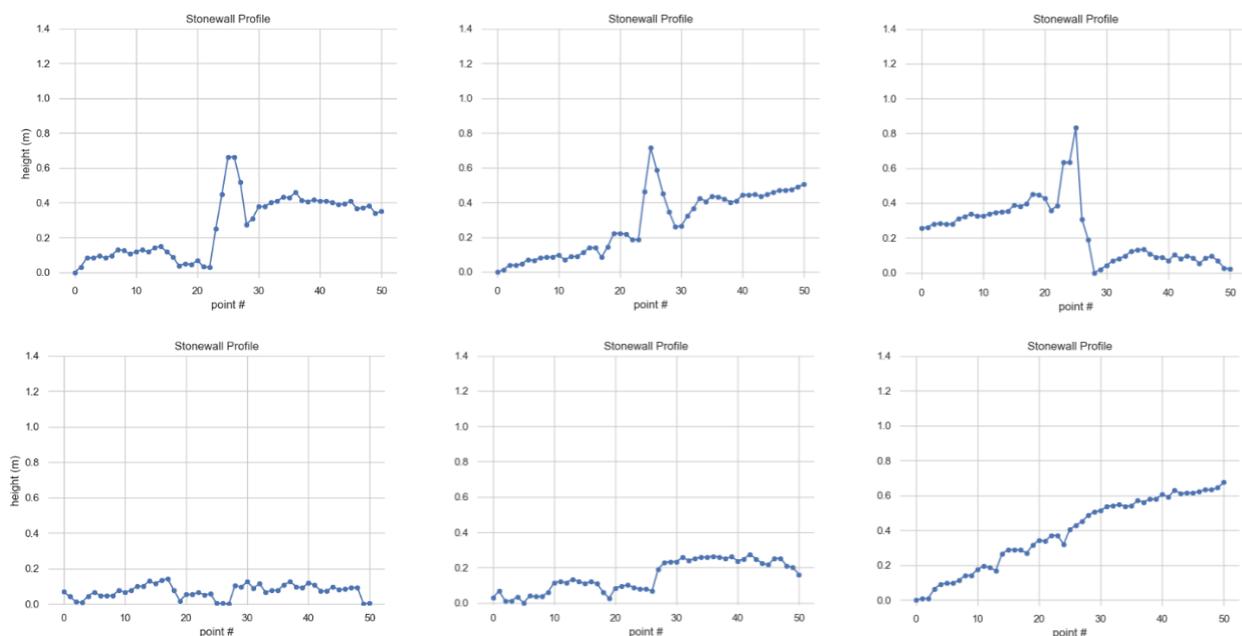


Fig. 7 - Examples of stone wall profiles. The top three profiles show the presence of stone walls, while the bottom three are examples of profiles where a wall is no longer present

Step 2: Identifying Stone walls

The datasheets provided by the Culture Ministry provide a reference of which dimensions a stone wall can have. Additionally, even by plotting for a with cursory visual inspection, it is clearly observable whether or not a profile represents a wall. With a small number of walls or profiles it would be possible to sort the profiles manually. However, due to the large amount of profiles created in Ærø alone, an automated method was necessary. The *'find peaks'* function of the SciPy Python package [34], originally intended for use in Computational Biology and Bioinformatics, was utilized to identify 'peaks', which in our case were the peaks of the walls. This method was extremely successful at identifying stone walls that had a prominent peak, which luckily was the case for the bulk of the dataset. However, walls that were damaged, partially removed or covered with earth over time, were more challenging for the algorithm to correctly categorize. Walls of such type were marked with an *'unclear'* category for later manual inspection, while the remaining walls were designated either the *'wall'* or *'not wall'* category.

Additionally, the *'find peaks'* function stored the index of each peak. This allowed for the calculation of the position of the peak in relation to the cross-section. In the event that multiple peaks were found, only the peak that was closest to the centre of the profile were stored. These peak indexes were used to adjust the position of the linestrings in the stone walls dataset to the stone walls actual position on the ground, in the following step.

Step 3: Rebuilding the Dataset

Firstly, it was necessary to recreate the dataset with only the sections that were categorized as wall. This was achieved by redrawing the linestrings after their categorization. For an individual segment of stone wall, starting at the first profile, a linestring was drawn between the peak of the current wall and the peak of the following wall. This was performed only in the case that the current and next wall were categorized as walls. In this way, the entire dataset was redrawn with the non-wall sections removed, and with the remaining walls adjusted to their actual position as reflected in the DTM. This provided for the feature engineering of the dataset, in order to maximize the extraction of features suited to represent the target data for the CNN model training [35].

Step 4: Validation

For the majority (>90%) of the profiles, the presence of a stone wall was easily identified. However, for the cases where the wall was either very low, or partially removed, the profiles could not be reliably classified by our simple algorithm. As described above, any profile that was not quickly identified (the prominence of the peak was less than 0.3m), a third class *'unclear'* was attributed. In lieu of developing a more robust and complex algorithm, it was necessary to manually inspect these profiles with a visualization software tool (QGIS [36]). This inspection was done by visually comparing the *'unclear'* profiles against the DTM in hillshade, coupled with the profile tool plugin³. Any profiles that suggested the wall no longer was present, or if verification was not possible using these methods, were removed from the dataset.

4.2 Profile Based Machine Learning

During the earlier stages of development, Deep Learning was used on the profiles to see if an improvement in profile classification could be observed. This was done in Python using Keras [37] with TensorFlow on the backend [38]. Deep Learning was done on elevation array of a profile (1x50 array of float values). The concept was to train a relatively shallow Deep Learning Model on the classified

³ Available at the repository <https://github.com/PANOimagen/profiletool>

walls, and use this model to classify the walls again. By doing this we hoped to see an improvement on our algorithm-only classification model. The problem was approached in the following two ways: 1) regression and 2) classification.

A regression-based algorithm used an integer label which represented the index location of the peak in the elevation array. The output of each input profile is a value that designates the peak index in the array, between 0 and 50 for a peak, and -1 for no peak.

A classification-based algorithm simply sorted the arrays into two categories. The same model architecture (Fig. 8) was used as in the regression problem, with changes to the loss function. Instead of each array being accompanied by the integer of the peak index, it was designated a 'wall' or 'no wall' category.

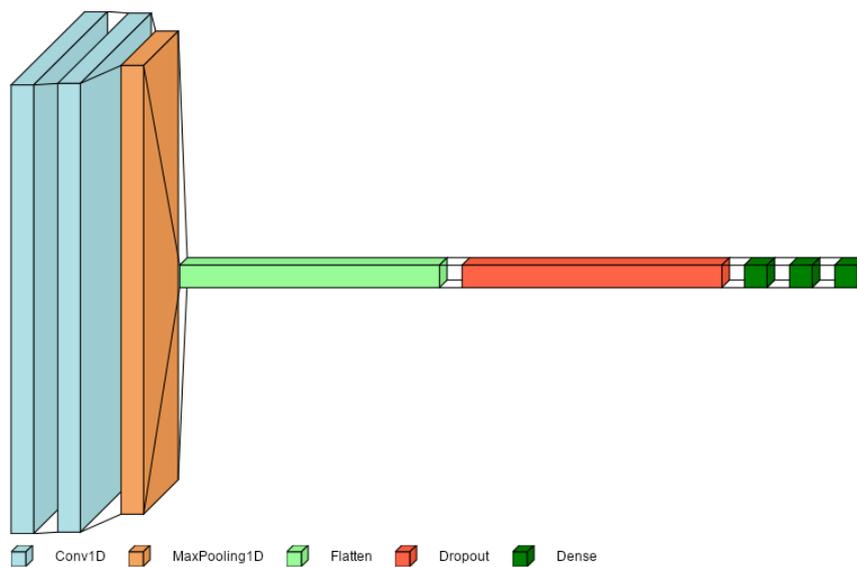


Fig. 8 - Classification model architecture used

After a period of experimentation, it was decided that the method did not yield significantly different results from the results where no DL was used. Additionally, the regression-based model was not effective at identifying the peak index. While there seemed to be an improvement on the classification of profiles using the classification-based method, it was difficult to quantify how much better given the input data itself had a degree of uncertainty. For this reason, neither of these methods were pursued further, and neither process was used in producing the final results.

4.3 Data Preparation and image patches generation

The next step was to train a CNN on the updated stone wall dataset. For this, the stone wall would first need to be converted into a raster format. The DEM data, which would be used as training data, had a spatial resolution of 0.4m per pixel, therefore the stone walls dataset would need to be rasterized to match this resolution. During the transformation to raster format, a down-sample of pixels was first performed (down-sampled to 10 cm), in order to expand the number of presence pixels. The pixels were then restored back to the target resolution of 0.4m, resulting in an anti-aliased walls dataset (example on Fig. 9). All the pixels covering the location of the wall were given a float value ($0 < x \leq 1$), commensurate to the distance from the centre of the pixel to the centre of the wall. Pixels which intersected the centreline of the wall were given a value 1.0. This transformation helped to account for spatial uncertainty, but also gave flexibility to the output prediction. In doing this, we turn the problem into one of regression, which we hoped would help to compensate for the number of absence pixels present in the patches, while extracting the target data for training the model.

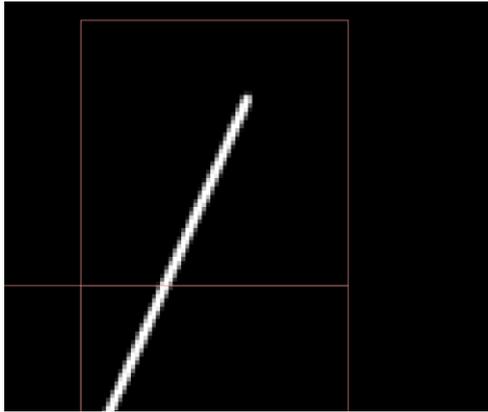


Fig. 9 - Example of a stone wall anti-aliased



Fig. 10 – Example the location and size of 64x64 pixel patches in the label data

Deep Learning models that employ CNNs require the training data be in the form of small patches, since the spatial context information is learned by filters. Additionally, labels must be provided along with associated environmental data. In our case, the rasterized stone walls acted as the labels, and the stacked DTM, HAT and Sobel filter were the associated image data. Using scripts produced in Python, the training data was broken into 64x64 pixel patches, which resulted in a total of 24.782 patches of stacked DEM data, and the same amount of associated rasterized stone wall labels (example on Fig. 10).

Given the size of our training data, and in order to increase generalizability and decrease overfitting within the model, Data Augmentation was used [39]. Data Augmentation is the process of increasing the amount of available data by adding manipulations to the ‘real’ dataset, to create additional data that is similar – this new data is called ‘synthetically modified data’. Specifically, we implemented a geometric transformation by rotation augmentation on each patch, such that each patch was rotated 0, 90, 180, and 270 degrees. In the case of our data, this method is considered as “safe” augmentation transformation, given its likelihood in preserving the label [39], because the assumption was that stone wall structures do not have a specific orientation. The rotation was applied to the data after splitting the dataset into train and test sets, applied only to the latter. Such transformations can help reduce overfitting by creating more training data [39].

For the image inputs, patches were extracted in the buffered area around the rasterized stone wall dataset. Due to the relationship between the buffer distance and the size of the patches (64x64 pixels), there were some patches which contained no stone wall at all. These types are patches are known as absence data. In order to investigate the effect this absence data had on our results, additional absence data was added to the final dataset, which was created by extracting patches from other areas on Ærø. Because it was initially theorized that the model would have most difficulty in urban areas and in differentiating modern walls from historic stone walls, 9.307 patches of absence data were added to the dataset from urban areas. This was done by analyzing the land cover of the island using a CORINE landcover type layer, and extracting the patches from urban areas where no protected walls were located. Once the absence data was added, the initial dataset of size 24782x64x64 was increased to size 33819x64x64.

When training the model, the dataset was shuffled and then split in train and test sets, in a ratio of 70-30% respectively, using Scikit-Learn’s function ‘train_test_split’ [40]. For the train set, after augmenting the data with rotations, the final size of the dataset was 94692x64x64.

The inherent imbalance in the dataset, where each patch contained many more ‘0’ (no wall) pixels, compared to the number of ‘1’ (wall) pixels, caused difficulties with the loss function. Mean Squared

Error (MSE) was used as the model’s loss function, and given its calculation method, the loss values ended up being very small. Due to difficulties with the model while using very small loss values, initial attempts at training the model resulted in failure, due to a complete inability of the model to learn. By multiplying the label patches by 1.000 (with a final target value between 0 and 1.000), we were able to solve this issue. The values of the labels on the output were then converted back to their original scale.

4.4 Model Training and Prediction

The creation and training of the DL model was done using the Python programming language. The Keras [37] and TensorFlow [38] packages were used to create the model.

For this step, A Fully Convolutional Neural Network (FCNN) model design was used, as this allowed for the output of a prediction raster of the same size as the input (64x64 pixels). The model used has a U-Net-like architecture, with initial down-sampling followed by up-sampling. The U-Net model has an expansive path symmetric to the contracting path, leading to a U-shaped architecture. After each down-sampling convolution (a 3x3 convolution followed by a rectified linear unit (ReLU) and a 2x2 max pooling layer of stride 2), a skip connection (concatenation of a corresponding layer between the contracting and expansive path) is performed, to provide information of localization accuracy, reduced by the use of max-pooling layers [23]. In this study, the model architecture followed a similar architecture to that of U-net, employing 6 down-sampling blocks, each containing two convolution layers of 3x3-sized filters, where each block was followed by a 2x2 max pooling layer of stride 2 and with ‘0’ padding. The three down-sampling blocks used 32, 64 and 96 filters each respectively. For the expansive path, two 3x3 transposed convolution layers were used, with 64 and 96 filters respectively, followed by a concatenation of the previous transposed layer with a layer from the expansive path (see Fig. 11 for a visualization of the model’s architecture). The model accepted input an input vector of size 64x64x3, one channel each for the layers DTM, HAT and DTM with Sobel filter. The output was a vector of size 64x64x1. A ReLU activation function was used in the final layer, where each individual pixel within the patch was designated with a value higher than 0, corresponding to the likelihood that each pixel contained a stone wall. Each pixel corresponds to an area of 0.4m² on the original raster.

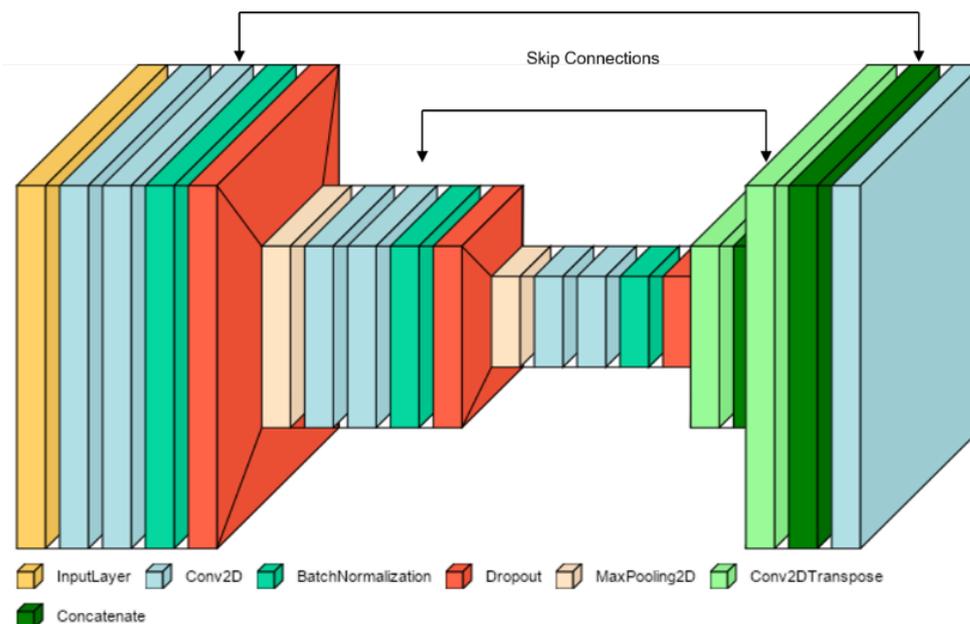


Fig. 11 - Model architecture of the CNN model with a shallower structure compared to the U-Net original model

Besides the lower number of blocks and filters used in both the contracting and expansive paths, the activation function used in the convolution layers differed from the original U-Net architecture. Instead of the ReLU function, a similar function named Swish [41] was applied⁴, after testing the model through various iterations with both, and the latter resulted in better overall model performance. The adaptive moment estimation (Adam) optimization was used [42], and the learning rate was initialized at 0.001. In order to optimize learning and have the model quickly converge to a good solution, a step decay for the learning rate was scheduled, whereby the learning rate was set to decrease by 0.5 after 5 epochs of similar loss results. Early stopping was also applied, by monitoring the validation loss function during training, where the training comes to an end when the loss is no longer decreasing. This is decided by setting the minimum change that qualifies as improving (patience), corresponding to the number of epochs after which the training will stop if there is no improvement (set to 5). The early stopping function was applied using Keras package callback *'EarlyStopping'*. The validation loss is calculated on the validation data, which is split from the training data (20%).

The optimization of hyperparameters is important in achieving the best possible results with a CNN [18] [17]. With a FCCN, the most significant being the number of filters used in each convolution layer, the size of the kernel that executes the filter, the size of the 'stride' as the kernel moves across the patch, and the padding for the cases where the kernel area goes off the edge of the patch. A complete grid search of the entire hyperparameter space was not possible due to time constraints, however after narrowing down to certain number parameters, the final set of values previously described was defined, after iterating the model 5 times, averaging the results and selecting the values with overall best performance. This evaluation method also includes the additional parameters previously described.

The loss function used was the Mean Squared Error (MSE), also known as L2. The MSE loss function minimizes the squared differences between the estimated and the target values, and is one of the most typically used for regression problems [43]. According to [44] study, the choice of a loss function to use is dependent on the on the dataset and on the model architecture, and it is recommended to try different losses at the initial stage of the process. During training, four different losses were used to run the model, namely the Mean Absolute Error (MAE), also known as L1, Huber loss, a smooth mean absolute error, log cosh, a smooth mean squared error, and MSE. We iterated the model training multiple times over the same loss function, and adapting the parameters in order to understand the impact of the loss in the results. Given that in the present work we do not attempt to make a comparison of the different possible loss functions to be used in a regression problem, we will not go deeper on this analysis, leaving it instead as a possibility for future development. However, from the various functions tried, only MSE provided a stable learning and usable results.

After model training, a prediction could be made using stacked DTM, HAT and Sobel filter data as inputs, outputting a series of patches representing the predicted areas of stone walls. The output was in the form of a 3-Dimensional array which was then converted to raster format. This was done using the already referenced Buteo Toolbox in Python. During the prediction, offsets were applied on the predicted patches in order to reduce the noise produced by the patches borders during the extraction of the images. A sequence of offsets were used: 1) 16x16, 2) 32x32 and 3) 48x48; merged with the median.

⁴ Swish was developed by Google Brain Team as an alternative to ReLU, where tends to work better in deeper models and on different datasets for image classification. For a more thorough explanation and comprehension of the function, see the reference paper [41].

4.5 Assessment and Post-Processing

The lack of studies referencing this stone walls dataset, as well as the dataset's inaccuracy and overall poor condition, present a serious challenge when comparing our results. Because of this, the results of our analysis are compared with original dataset, where the assumption is made that the locations of the stone walls here are ground truth.

The assessment of the results was undertaken using a combination of quantitative and qualitative analysis. During quantitative assessment analysis, the performance and results from the loss function (MSE), additional metrics (MAE and RMSE), and the validation loss were analysed. This was done after iterating the model multiple times and averaging the results. An assessment based on the pixel value of the prediction was done by comparing the true values and the predicted values for each patch. Taking the intersection of the presence pixels in the input and prediction, as well as the intersection of the absence pixels, and dividing by the number of pixels, giving a number between 0 and 1, a representation of the overall accuracy of the prediction was calculated.

In order to obtain a more accurate assessment of the predictions on the true values, a small validation area was selected. This area, as best as could be ascertained given the dataset, reflected the ground truth. The values of the pixels for the selected area 'ground truth' were compared with the predicted values using the same pixel-wise metric.

The qualitative analysis was focused on the assessment based on the visualization of the predicted images. The results from the different training runs were inspected, comparing with the original stone walls dataset, to verify for: 1) the prediction of stone walls in relation to the existing structures; 2) new predicted walls; 3) walls or segments of walls that were no longer present, therefore not predicted; 4) prediction errors and false positives; 5) general noise present in the predictions, surrounding the predicted values and the absence areas. As a final assessment, a field inspection was conducted, by first selecting cases of stone walls that would be relevant to verify. In this way, a verification of predicted stone walls and comparison with the ground truth was done in order to provide additional validation.

The output of the model shows predicted wall locations in the study area, where pixels with a value > 0 indicated the presence of a stone wall. The final step of the analysis is to separate the walls into their respective categories: 1) walls that appear both in the initial dataset and the prediction 2) walls that appear in the initial dataset and not in the prediction, 3) walls that are not present in the initial dataset but do appear in the prediction.

The method chosen to do such analysis was to compare the original dataset with the final prediction, and to observe the differences between the two datasets. Removing walls that appear in both the prediction and the original wall dataset from the prediction raster reveals the new walls that have been 'discovered' by the algorithm. Removing walls that appear in both the original dataset and the prediction from the original dataset raster leaves only the walls that have been 'removed'.

Because of the discrepancies in wall placement between the original data and the prediction raster, it was unfortunately not possible to use simple raster algebra ($\text{original} - \text{prediction} = \text{result}$). Instead it was necessary to create a proximity raster, and use this raster to delete any wall that was within 25pixels (10m) distance of the wall. This led to the final '*found*' and '*removed*' walls being represented as shorter than they were in reality, however it gave the clearest and simplest to interpret results. Lastly, a field visit was conducted to the study area, in order to validate some of the predictions obtained.

5. Results

5.1 Quantitative Assessment

Fig. 12 shows the loss results per epoch, averaged from after 5 iterations of the model. The average run had 21 epochs with early stopping (set to run for 50 epochs), which monitored the validation loss (the loss calculated for the validation data), after reaching a maximum of 5 epochs without improvement on its value.

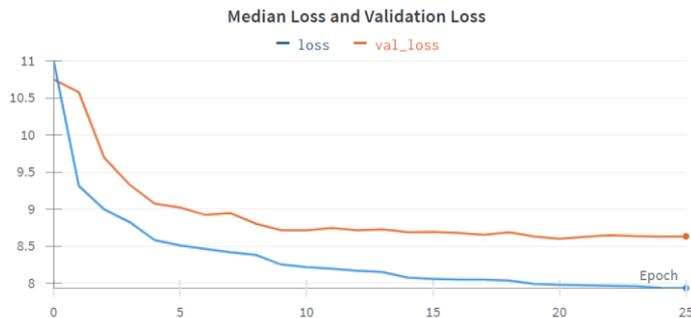


Fig. 12 - Loss and validation loss with calculated median for all the final model iterations

	Train	Validation	Test (Evaluation)
MSE	7,976	8,642	7,294
MAE	0,018	0,017	0,016
RMSE	0,089	0,092	0,085

Table 1 - Summary statistics of the model for training, validation and evaluation (test) data

The evaluation on the test data shows a MSE value of 7,29 (Table 1), indicating the summed error for all pixels that compose one patch. This was useful to compare between the iterations of the model, as well as during the testing of other additional attempts, alterations and improvements to the model. Since comparable data or previous research is not existent, to the best of our knowledge, we are not able to compare such evaluation results with reference values. These values were used to measure the results from constant optimization and improvement of the model. The results from the model training with using only the DTM, or DTM and Height above Terrain (HAT) were slightly similar in comparison to the ones obtained with the three layers, where the evaluation loss and MAE had higher values (Table 2).

	DTM	DTM & HAT	DTM, Sobel & HAT Without Absence data	DTM, Sobel & HAT
Loss (MSE)	6,329	6,414	11,029	7,294
MAE	0,015	0,014	0,024	0,016
RMSE	0,079	0,080	0,105	0,085
Pixel-wise metric	0,86	0,88	0,83	0,88

Table 2 - Evaluation metric results for different training datasets

With more significance in the evaluation metrics' values, the inclusion of absence data provided improvements to the output results and significantly lower loss and MAE values. The absence data added represented the areas where stone walls are not supposed to be present (mainly representing urban areas). The difference between the pixel values predicted and the true pixels representing a stone wall for the validation area also changed with the different data layers used in the prediction model. The addition of the Sobel filter and the HAT increased the similarity on the number of pixels representing stone walls, during the prediction on the test data. The final model add an average of 0.88 for the pixel-wise metric, corresponding to 88% of overall matching of number of pixels between predicted and true data. However, such numbers are to be considered with some reservation, since it does not provide much knowledge on the accuracy in the prediction of existent structures, or of new walls, despite shedding a light on differentiating the performance results of the different model runs.

More significantly, the pixel-wise metric calculated for the validation area, from the overall prediction of the municipality of Ærø, was of 0.93 for the final model, indicating a high prediction performance.

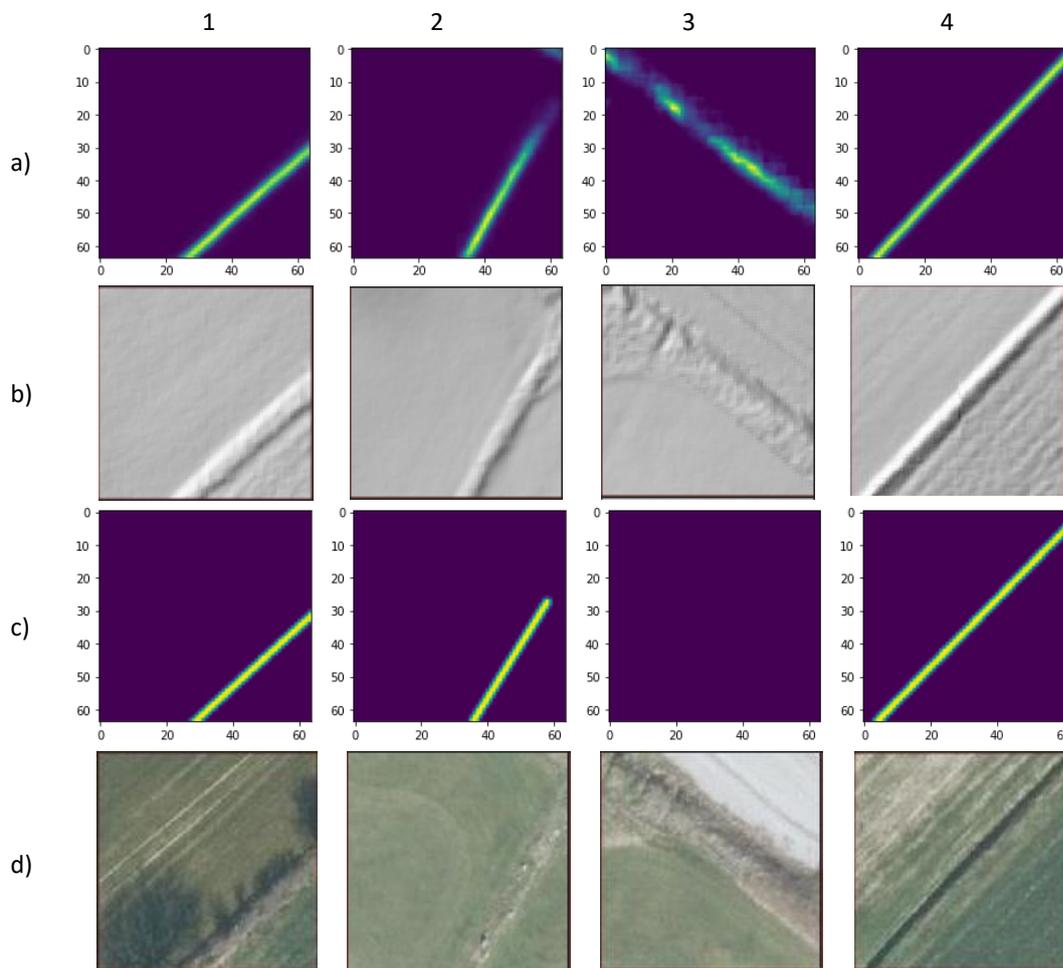


Fig. 13 - Example of patches of stone wall predictions and respective labels and ground truth, where column 1, 2 and 4 are example of correctly predicted stone walls, and column 3, a suspiciously predicted stone wall. Horizontally: a) prediction patches; b) dtm in hillshade; c) stone wall dataset (label); d) aerial image from kortforsyning.dk (Spring 2020).

The output of the predictions on the evaluation data (test dataset) show a clear picture of the ability of the terrain data to identify stone walls on the landscape. Fig. 11 Fig. 13 shows some examples of predicted stone walls (first row (a)), where image 1 and 4 display a clear detection of stone walls, as confirmed by the ground truth data (rows b and d) and stone walls dataset (row c). On column 3, an example of an unsure result of a stone wall prediction is displayed, where no stone wall is existent in the dataset, and the aerial image shows on what might be a ridge, in need of further validation.

5.2 Qualitative Assessment and External Validation

The final predictions for the municipality of Ærø showed positive results for both the identification of removed segments and walls, and new stone walls. The visualization of the final predictions, suggests that the model can generally identify stone walls, corresponding to the stone wall dataset, even in areas with more dense vegetation. However, for walls located in areas of dense forest, or wall structures that have very little salience, some errors of false negative predictions can occur (Fig. 14).



Fig. 14 - Examples of model predictions: Top left, shows an example of the challenge in predicting on dense forested areas, and top right, on shallower structures (on the top corner to the right, an error of prediction can be detected); Bottom left, an example of an unregistered wall predicted, and a wall that seems to have been removed; and bottom right, examples of registered as well as unregistered walls predicted.

For the models trained only on the DTM, or DTM and HAT layers, it is possible to identify the differences on the predictions. For the first one, a lower ability to identify the stone walls was detected, especially in differentiating amongst different types of edges, while for the latter, the level of noise, composed of low pixel values was considerable (Fig. 16). A considerable difference is also visible for the predictions that included absence data, where much of the noise and scattered pixels were removed (Fig. 15).



Fig. 15 - Difference between prediction with absence data (left) and without (right)



Fig. 16 - Difference of predictions between using DTM alone (left), DTM and HAT (center), and the final model (DTM, Sobel filter and HAT) (right). In yellow: stone walls dataset.

In order to test the generalization of the model, an external validation was done, by predicting for a new dataset. The municipality of Silkeborg in the region of Jutland, has a more diversified landscape, and has a larger extent.

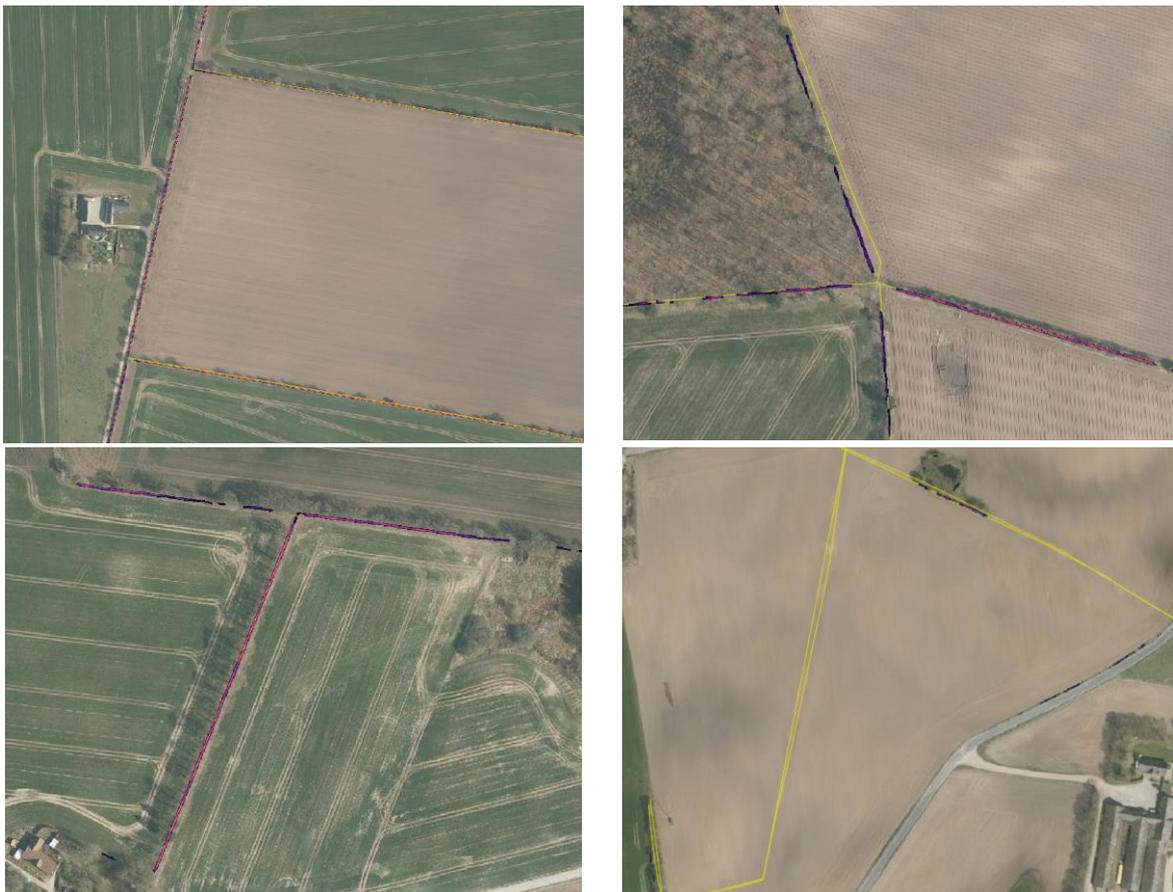


Fig. 17 - Example of External validation predictions in Silkeborg: Top, left shows the prediction of registered walls and a possible unregistered wall; Top, right the prediction follows the stone walls registry; Bottom, left possible unregistered stone walls are clearly predicted; Bottom, right shows where probable nonexistent registered walls that were not predicted as such.

On a visual inspection, the prediction shows an overall good generalization of the model, where the stone walls are clearly distinguished. New stone walls seem also to appear on the prediction, some

with a high level of definition, suggesting a good application of the model for detecting new potential stone walls in Denmark (Fig. 17).

5.3 Postprocessing

An initial postprocessing method was done, in order to reduce the noise created by the edges of the patches, by applying offsets during the prediction. Such method allowed to create clearer prediction images and eliminate unrelated values. For the visualization of the predictions, a visual scale was applied, where first values between 0 and 1 would be considered (corresponding to the reference values of stone walls), but ending up to include only values above 0.2. Most of the values below this threshold were shown to be mostly noise values.

The final step of postprocessing analysis, which highlighted the 'removed' and 'found' stone walls from the final prediction, showed that, around 391 stone walls (Fig. 18) were flagged with either segments or entire removed walls (a total of around 37 km). These were mostly located on agriculture areas, where it is possible to identify whole segments of wall removed on crop fields, as well as segments on the edges of walls, for perhaps accessibility purposes.

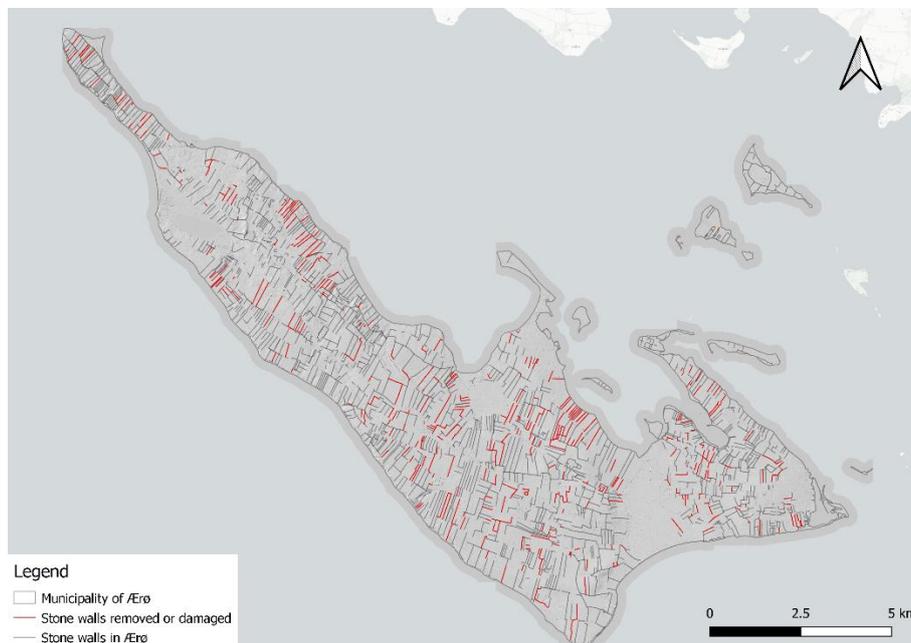


Fig. 18 - Stone walls predicted as removed or damaged after postprocessing

Many new stone walls were found in the prediction for Ærø, where varying sizes and definition could be identified. Such structures were then filtered by their pixel value and length, where only structures with values above 0.50 were considered, and smaller structures with less than 10 meters, were excluded (Fig. 19).



Fig. 19 - Examples of post-processed found walls in Ærø

For a final validation, a field visit was conducted to the study site, and in this way evaluate our results with ground truth. It was possible to verify some of the predictions, namely the removal of edges of stone walls, confirmed by the visit on-site. An entire wall that was identified as removed by our predictions was also verified and confirmed. Verification on-site allowed also to confront the predictions of unregistered stone walls, confirming in at least some of the cases, the existence of similar structures to protected stone walls, indicating the positive output of the prediction results (Fig. 20).



Fig. 20 – Onsite Photo 1: Hidden wall structure, where it is possible to observe the untouched primary composed by stones and rocks. This structure was predicted as stone wall, and it is not included in the registry.



Fig. 21 – On-site Photo 2: The model indicates a stone wall here running parallel to a sealed section of road. This transpired to be a small earthen embankment. These sections of predicted wall span the length of the study area and indicate a systemic inaccuracy in the prediction.

Additionally, errors of prediction were also verified, and the validation on-site allowed to perceive the nature of the visited predicted stone walls. This included structures identified as unregistered stone walls, but were in fact embankments and/or ridges (Fig. 21). A more detailed description of the field visit validation, see Annex 1 – Field site validation on p. 36.

The final results of the analysis can be seen here displayed in a prototype of a WebGIS visualisation tool. This tool was created using Leaflet [45] for JavaScript (Fig. 22). The development of such tool can be improved to include visualization options, namely the number of unregistered stone walls predicted, the number of segments and walls that have been predicted as removed, as well as the metadata necessary to identify the stone wall and update the registry.

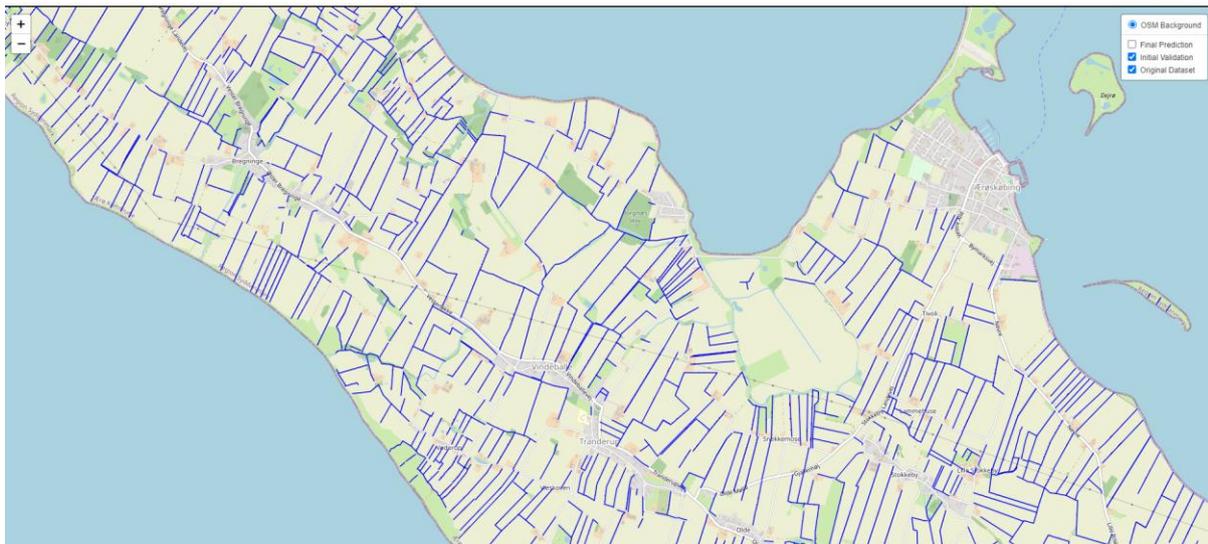


Fig. 22 - Example of a prototype for a WebGIS to visualize the results of the predictions

6. Discussion

Our results suggest that LiDAR derived digital elevation data can be used to extract terrain features from digital elevation data, as also found by [9] [25] [8]. Our first step sought to validate the stone walls dataset, by comparing them with the DTM and analysing their elevation profile. Interestingly, this method proved successful in identifying protected stone walls or segments of walls that no longer are present in the landscape, but are still registered, which was easily validated by inspecting the aerial images and elevation model to confirm the results. This method was applied as feature engineering for the Deep Learning step in the project, preparing the proper target data to be compatible with the model training [35].

The application of Deep Learning techniques, specifically the use of Convolutional Neural Networks on digital elevation data presented good results in identifying specific pixels where stone walls are present. Pixel-wise based analysis of the output predictions in the validation area suggested a high level of regression based accuracy (0.93), and on the overall area, an average of 0.88. The algorithm discovered areas both where the existent dataset needed to be updated due to the removal of stone walls and discovered new stone walls. These discrepancies seemed to be correct with the results from the first step analysis, as well as manual verification on location, where a selected number of new predicted stone walls presented similar characteristics as the ones already registered, however this remains to be confirmed by an expert in stone walls. Additionally, the postprocessing of the predictions identified a total of 391 stone walls removed or having segments that were removed/taken down, in Århus.

The use of multiple data sources showed an improvement over using the DTM alone. The results of the model improved with the inclusion of additional data layers, with the best model being trained on a combination of the DTM, HAT, and DTM(Sobel) layers. The visual inspection showed a decrease of noisy pixels on detected edges that are not stone walls, especially in urban areas. Future studies could also look at the effects of including supplementary data such as aerial imagery, historical maps, or the locations of municipality borders, where the presence of walls is most likely. Given the relation of stone walls with the presence of vegetation (where some structures have vegetation and even trees

located on top), the use of NDVI imagery could potentially yield interesting results. In fact, a future study should relate the presence of stone walls with a biodiversity measurement such as the bioscore [46], relevant not only for structures detection, but also engage in the promotion of their value in biodiversity conservation in areas of persistent habitat fragmentation due to agricultural development.

The results of this analysis could be compared to a raster-based analysis where the Sobel filter or comparative edge detection techniques could be used. Such approach was initially considered, but dismissed after first analysis. The difficulty with this approach is that it is far less discriminating, identifying all of the images present in the image, making necessary to sort then differentiate the walls from other edge-type objects. For this specific case, it showed that additional spatial context is needed in order to distinguish amongst the different structures. The findings of this study show a significant improvement on the edge-detection method, because the algorithm is able to distinguish a stone wall apart from most edge-like structures. However, given the unique nature and context of our dataset, we are not able to relate our findings with other studies approaching the same problem, nevertheless our findings do support those of [7] and [25], who also explored the value of CNNs for extracting features from digital terrain data.

The dataset itself diverges from classical deep learning problems, in that the training data itself is not completely validated. Given the data science adage “garbage in garbage out”, it is paramount in all machine learning tasks that the training data is correctly labelled. In the case of our study, the original dataset from which the study is based, is not actually representative of ground truth. The first step of our analysis sought to curb the effects of this issue, by removing as many walls as possible from the dataset that were either absent, or dubious. Additionally, given that the task was to find new walls within the dataset, it is notable that our aim was not to achieve the lowest possible loss value, as perfect agreement between the prediction and the test data would suggest no new walls, which we knew not to be the case.

The advantage of analysis of DEM data is that the algorithm is able to identify patterns that are not visible using aerial or satellite imagery. This useful in archaeological applications for example, where vegetation cover can be an issue [8].

There are some limitations associated with our study. Firstly, we are limited by the availability of our terrain data. This data is available for the entirety of Denmark, but is only released in connection with a new aerial LiDAR mission, which currently is renewed only every 7 years. The data used in this study was from 2014, and therefore is unable to detect and map changes that have occurred in the intervening time. The results can also highly depend on the quality of the LiDAR derived data, where point cloud densities can influence the ability to detect and correctly identify small and narrow objects and structures, as mentioned by [47]. In order to apply the same analysis on an updated product from a new LiDAR mission, such differences and consequent biases need to be considered. Additionally, while terrain data is available for Denmark, it is not necessarily available in other locations, nor with the same characteristics. Although the application of this study can be generalized to the context of other countries with similar protected structures, such as Sweden, Norway or Scotland, it is dependent on the level of resolution of their available national LiDAR data, a relevant component, as confirmed by [47].

Due to length computational time, it was not possible to fully optimise the CNN method for the task of identifying stone walls. Exhaustively testing the model for the best possible hyperparameters and architecture was not practical given time constraints and computer power, and was also outside the scope of this study. In either case: 1) all findings would probably require validation by an expert in the

field, 2) small increases in model accuracy were unlikely to result in the discovery of new walls, only that the walls extent would be slightly more accurate. If this method were to be utilized for a similar task, it may be worth experimenting with other model architectures such as Mask R-CNN, as has been used by [25], or ResNet, as in [7].

Some difficulties were encountered in applying the CNN performing a regression task, given the challenge in evaluating the results. Considering that the target data presented an high imbalance between presence and absence data for each patch extracted, a classification into wall/no-wall proved to be less stable while running the same model with a classified output (where the last layer activation function was switched to Sigmoid, and the loss function to binary cross entropy). Therefore, a deep regression was applied for this specific problem, especially given that it was relevant to analyse and predict with continue values representing the probability of a protected stone wall to be present or not. Nonetheless, the regression-based method produces results which are harder to interpret and compare. Furthermore, to the best of our knowledge, there are no previous studies on automatically identifying stone walls, that would enable for a comparison of performance or results, where the only reference source is the stone walls dataset itself.

The application of this study is undoubtably of relevance to the municipalities of Denmark. It can contribute to the automatization of the identification and update of the stone walls' registry, and in this way fulfil the recommendations outlined by the Ministry of Culture in Denmark [4]. The development of such tool can come in a shape of a Decision Support System, where each municipality could visualize and apply analysis on their specific dataset, and in this way, contributing to the update of the national registry. A prototype is already in development, and will also require the involvement of experts, and feedback of its usability by municipalities. Additionally, it would also be interesting to consider the benefits that citizen science can offer, whereby citizens could contribute to the updating of the stone walls registry by providing information on the field, similar to other nature-related applications (*GBIF* or *iNaturalist*)⁵.

7. Conclusion

This study shows the value of using a CNN Deep regression for extracting features from Digital Elevation Data, by attempting to map stone walls in a study site in Denmark, and in this way update their registry. We used publicly available data and concentrated on the Danish municipalities of Ærø, and Silkeborg (for external validation). Using pixel-wise evaluation, there was an overall agreement of 93% between ground truth and prediction of stone walls in the validation area. Good results were seen using the DTM alone, however better results were obtained when adding HAT and an additional DTM layer with a Sobel filter applied. Good generalizability was found when externally validating the model on new data, showing good results for either the existent stone walls, as well as predicting new potential ones. The method performed best in open areas, however positive results were also seen in forested areas, which suggests that this method could be useful in the identification of features that might be challenging to detect, using remote sensing techniques alone. In order to further improve the identification of stone walls, we suggest that the inclusion of a multi-modal dataset could be beneficial, in order to add additional context and improve differentiation. Further improvements can also be made by exploring the methodology and optimization of the deep learning CNN model.

Recent developments in Machine Learning and CNN's, and the increasing availability of LiDAR-based terrain models is an exciting development, which facilitates new methodologies in the extraction and mapping of terrain features, structures and objects. This application demonstrates the usefulness in

⁵ <https://www.gbif.org/> and <https://www.inaturalist.org/>

using such methods, with the potential to not only automatize processes in local and land management, but also to engage in innovative analysis in order to find scalable solutions for relevant problems.

The code used in this study can be found in the associated GitHub repository (<https://github.com/AnaCMFernandes/stonewalls>).

8. References

- [1] K. Kulturministeriet, "Vejledning om beskyttede sten- og jorddiger," Kulturarvsstyrelsen, Copenhagen, 2009.
- [2] J. Pedersen, "Jord- og stendiger," 2021. [Online]. Available: <https://aktiv.dn.dk/sagsarbejde/natur/jord-og-stendiger/>.
- [3] Kulturministeriet, "Bekendtgørelse om beskyttede sten- og jorddiger og lignende, BEK nr 1029 af 21/10/2004," 21 October 2004. [Online]. Available: <https://www.retsinformation.dk/eli/lta/2004/1029>.
- [4] H. A. Christensen, "Nyt forsøg: Kulturminister vil kortlægge beskyttede sten- og jorddiger," 1 December 2020. [Online]. Available: <https://landbrugsavisen.dk/nyt-fors%C3%B8g-kulturminister-vil-kortl%C3%A6gge-beskyttede-sten-og-jorddiger>.
- [5] SDFE, "DTK/4-cm kort (1977-1992)," SDFE, Kortforsyning, 1977-1992.
- [6] Z. Chen, B. Gao and B. Devereux, "State-of-the-Art: DTM Generation Using Airborne LIDAR Data.," *Sensors*, vol. 17, no. 1, p. 150, 2017.
- [7] T. ØD, C. DC, Waldeland and AU, "Using deep neural networks on airborne laser scanningdata: Results from a case study of semi-automatic mapping of archaeological topography on Arran, Scotland.," *Archaeological Prospection*, vol. 26, p. 165–175, 2019.
- [8] A. F. Chase, D. Z. Chase, C. T. Fisher, S. J. Leisz and J. F. Weishampel, "Geospatial revolution in Mesoamerican archaeology," *Proceedings of the National Academy of Sciences*, vol. 109 (32), pp. 12916-12921, 2012.
- [9] A. Guyot, M. Lennon, T. Lorho and L. Hubert-Moy, "Combined Detection and Segmentation of Archeological Structures from LiDAR Data Using a Deep Learning Approach," *Journal of Computer Applications in Archaeology*, vol. 4, no. 1, pp. 1-19, 2021.
- [10] The Earth Archive, "The Earth Archive," [Online]. Available: <https://www.theeartharchive.com/>. [Accessed 10 May 2021].
- [11] J. E. Moeslund, L. Arge, P. K. Bøcher, T. Dalgaard, M. V. Odgaard, B. Nygaard and a. J.-C. Svenning, "Topographically controlled soil moisture is the primary driver of local vegetation patterns across a lowland region," *Ecosphere*, vol. 4, no. 7, pp. 1-26, 2013.
- [12] A. U. Waldeland, A. -B. Salberg, T. Ø. D. and A. Vollrath, "Large-Scale Vegetation Height Mapping from Sentinel Data Using Deep Learning," in *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, 2020.
- [13] J. E. Moeslund, A. Zlinszky, R. Ejrnæs, A. K. Brunbjerg, P. K. Bøcher, J.-C. Svenning and S. and Normand, "Light detection and ranging explains diversity of plants, fungi, lichens, and bryophytes across multiple habitats and large geographic extent," *Ecological Applications*, vol. 29, no. 5, p. e01907, 2019.

- [14] N. Muhadi, A. Abdullah, S. Bejo, M. Mahadi and A. Mijic, "The Use of LiDAR-Derived DEM in Flood Applications: A Review," *Remote Sensing*, vol. 12, no. 14, p. 2308, 2020.
- [15] J. Im and J. R. J. & M. E. Hodgson, "Object-Based Land Cover Classification Using High-Posting-Density LiDAR Data," *GIScience & Remote Sensing*, vol. 45, no. 2, pp. 209-228, 2008.
- [16] W. Y. Yan, A. Shaker and N. El-Ashmawy, "Urban land cover classification using airborne LiDAR data: A review," *Remote Sensing of Environment*, vol. 158, pp. 295-31, 2015.
- [17] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, Cambridge: MIT Press, 2016.
- [18] Y. B. Y. & H. G. LeCun, "Deep learning," *Nature*, no. 521, p. 436-444, 2015.
- [19] G. J. Scott, M. R. England, W. A. Starms, R. A. Marcum and C. H. Davis, "Training Deep Convolutional Neural Networks for Land-Cover Classification of High-Resolution Imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 4, pp. 549-553, 2017.
- [20] L. Ma, Y. Liu, Z. Xueliang, Y. Yuanxin, Y. Gaofei and B. A. Johnson, "Deep learning in remote sensing applications: A meta-analysis and review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 152, 2019.
- [21] A. Sharma, X. Liu, X. Yang and D. Shi, "A patch-based convolutional neural network for remote sensing image classification," *Neural Networks*, vol. 95, pp. 19-28, 2017.
- [22] J. Long, E. Shelhamer and T. Darrel, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Trans Pattern Anal Mach Intell.*, vol. 39, no. 4, pp. 640-651, 2017.
- [23] O. Ronneberger, P. Fischer and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Munich, 2015.
- [24] A. E. Maxwell, M. S. Bester, L. A. Guillen, C. A. Ramezan, D. J. Carpinello, Y. Fan, F. M. Hartley, S. M. Maynard and J. L. Pyron, "Semantic Segmentation Deep Learning for Extracting Surface Mine Extents from Historic Topographic Maps," *Remote Sens.*, vol. 12, no. 24, p. 4145, 2020.
- [25] A. E. Maxwell, P. Pourmohammadi and J. D. Poyner, "Mapping the Topographic Features of Mining-Related Valley Fills Using Mask R-CNN Deep Learning and Digital Elevation Data," *Remote Sens*, vol. 12, no. 3, p. 547, 2020.
- [26] N. Sambasivan, S. Kapania, H. Highfill, D. Akrong, P. Paritosh and L. M. Aroyo, "Everyone Wants to Do the Model Work, Not the Data Work": Data Cascades in High-Stakes AI," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, Yokohama, 2021.
- [27] SDFE, "DHM/Terræn," SDFE, Styrelsen for Dataforsyning og Effektivisering, 2014.
- [28] SDFE, "DHM/Overflade," SDFE Styrelsen for Dataforsyning og Effektivisering, 2014.
- [29] SDFE, "Danmarks Højdemodel," SDFE, 2020. [Online]. Available: <https://sdfе.dk/hent-data/danmarks-hoejdemodel>. [Accessed April 2021].

- [30] O. R. Vincent and O. Folorunso, "A descriptive algorithm for sobel image edge detection," in *Proceedings of Informing Science & IT Education Conference*, Macon, GA, 2009.
- [31] S.-. o. Kulturstyrelsen, "Beskyttede sten- og jorddiger," Miljøportal, 2011.
- [32] K. M. -. S. o. Kulturstyrelsen, "Sten- og Jorddiger - Konkret udpegning," 30 July 2020. [Online]. Available: <https://slks.dk/omraader/kulturarv/arkaeologi-fortidsminder-og-diger/sten-og-jorddiger/konkret-udpegning/>. [Accessed March 2021].
- [33] SDFE, "Copernicus Land - corine land cover DNK (CLC)," Styrelsen for Dataforsyning og Effektivisering, 2012.
- [34] P. Virtanen, R. Gommers, T. Oliphant and e. al, "SciPy 1.0: fundamental algorithms for scientific computing in Python.," *Nat Methods*, vol. 17, p. 261–272, 2020.
- [35] He, K. Zhao and X. Chu, "AutoML: A survey of the state-of-the-art," *Knowledge-Based Systems*, vol. 212, pp. 1-27, 2021.
- [36] QGIS.org, "QGIS 3.16. Geographic Information System Developers Manual," QGIS Association, 2021. [Online]. Available: https://docs.qgis.org/3.16/en/docs/developers_guide/index.html.
- [37] F. Chollet and o. from, "Keras," GitHub, 2015.
- [38] M. Abadi, P. Barham, J. Chen, C. Z. D. A. and D. J. , "Tensorflow: A system for large-scale machine learning.," in *12th Symposium on Operating Systems Design and Implementation*, 2016.
- [39] C. Shorten and T. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J Big Data*, vol. 6, no. 60, pp. 1-48, 2019.
- [40] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel and J. Vanderplas, "Scikit-learn: Machine learning in Python," *J. Mach. Learning Res.*, vol. 12, pp. 2825-2830, 2012.
- [41] P. Ramachandran, B. Zoph and Q. V. Le, "Searching for Activation Functions," in *ICLR*, Vancouver, 2018.
- [42] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *ICLR*, San Diego, 2015.
- [43] M. Carvalho, B. Le Saux, P. Trouvé-Peloux, A. Almansa and F. Champagnat, "On regression losses for deep depth estimation," in *ICIP 2018*, Athens, 2018.
- [44] S. Lathuilière, P. Mesejo, X. Alameda-Pineda and R. Horaud, "A Comprehensive Analysis of Deep Regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 9, pp. 2065-2081, 2020.
- [45] V. Agafonkin, "Leaflet," 2021.
- [46] R. Ejrnæs, J. Bladt, J. Moeslund, A. K. Brunbjerg and G. B. Groom, "Biodiversitetskortets bioscore," 1 February 2018. [Online]. Available: https://dce.au.dk/fileadmin/dce.au.dk/Udgivelser/Notater_2018/Biodiversitetskortets_bioscore.pdf. [Accessed May 2021].

[47] I. Angelidis, G. Levin, R. Díaz-Varela and R. Malinowski, "Assessment of changes in formations of non-forest woody vegetation in southern Denmark based on airborne LiDAR.," *Environmental Monitoring and Assessment*, vol. 189, p. 437, 2017.

9. Annex 1 – Field site validation

The following images detail the examples of stone walls verifications on site, marked by their corresponding location in the study area.



Fig. 23 - On site photos for location 1, with corresponding prediction and orthophoto image on background (source: Kortforsyning)

- 1) The model indicates that the section of these walls close to the road have been removed. A on-site inspection indicates that this is the case, and that the approximate length of the removed sections is also reflected correctly in our prediction.



Fig. 24 - On site photos for location 2, with corresponding prediction and orthophoto image on background
(source: Kortforsyning)

- 2) The model indicates a long section of new stone wall dividing a residential block from its neighbouring field. An on-site inspection indicates that this is the case, and that the wall spans the entire length of the field. Additionally, the prediction also indicates the presence of a new stone wall on the northern border of the field, which was also confirmed on-site, and also spans the entire length of the field.



Fig. 25 - On site photos for location 3, with corresponding prediction and orthophoto image on background
(source: kortforsyning)

- 3) The model indicates that an entire stone wall has been removed which originally stood as the border between two fields. At the time of on-site inspection a crop of rapeseed had been planted, making it difficult to ascertain the wall's exact status, though it did appear to be removed. Aerial photography of the area seems to confirm this.



Fig. 26 - On site photos for location 4, with corresponding prediction and orthophoto image on background
(source: kortforsyning)

- 4) The model indicates a long section of new stone wall dividing two neighbouring fields. On-site inspection indicates that this is the case. The stone wall spans the entire length of both fields.



Fig. 27 - On site photos for location 5, with corresponding prediction and orthophoto image on background
(source: kortforsyning)

- 5) The model indicates two parallel sections of new wall. On-site inspection indicates that this is not the case, rather than the model has falsely identified two embankments running parallel to an unsealed road.



*Fig. 28 - On site photos for location 6, with corresponding prediction and orthophoto image on background
(source: kortforsyning)*

- 6) The model indicates several sections of stone wall in a large area dividing fields adjacent to a farm house. These walls are located on private land, so the presence of all of them could not be confirmed on-site, however inspection seems to strongly indicate that all of the walls predicted in this area are new stone walls.