

RESEARCH ARTICLE

Learning to Look at LiDAR: The Use of R-CNN in the Automated Detection of Archaeological Objects in LiDAR Data from the Netherlands

Wouter Baernd Verschoof-van der Vaart and Karsten Lambers

Computer-aided methods for the automatic detection of archaeological objects are needed to cope with the ever-growing set of largely digital and easily available remotely sensed data. In this paper, a promising new technique for the automated detection of multiple classes of archaeological objects in LiDAR data is presented. This technique is based on R-CNNs (Regions-based Convolutional Neural Networks). Unlike normal CNNs, which classify the entire input image, R-CNNs address the problem of object detection, which requires correctly localising and classifying (multiple) objects within a larger image. We have incorporated this technique into a workflow, which enables the preprocessing of LiDAR data into the required data format and the conversion of the results of the object detection into geographical data, usable in a GIS environment. The proposed technique has been trained and tested on LiDAR data gathered from the central part of the Netherlands. This area contains a multitude of archaeological objects, including prehistoric barrows and Celtic fields. The initial experiments show that we are able to automatically detect and categorise these two types of archaeological objects and thus proof the added value of this technique.

Keywords: Remote sensing; Object detection; R-CNN; Machine learning

1 Introduction

Generally, the data from remote sensing surveys is screened manually in archaeology. However, constant monitoring of the earth's surface—by a multitude of airborne and satellite sensors—causes a huge influx of data of high complexity and high quality. To cope with this ever-growing set of largely digital and easily available data, computer-aided methods for the processing of data and the detection of archaeological objects¹ are needed (Bennett, Cowley & De Laet 2014: 896).

Over a decade ago, archaeologists started developing computational methods for the (semi-)automated detection of archaeological objects (De Boer 2007; De Laet, Paulissen & Waelkens 2007). Since then multiple case studies have shown these algorithms to be capable of detecting well-defined archaeological traces, such as barrows (see for example Sevara et al. 2016). However, these (often) handcrafted algorithms are highly specialised on specific, single object categories and data sources, which restricts their use in different contexts and limits their usability in general for archaeological prospection. Furthermore, these approaches are predominantly complex algorithms that can require a high level of expertise

to operate, and are regularly dependent on expensive software. All this results in an implementation that is limited in its user-friendliness (see also Ball, Anderson & Seng Chan 2017: 3).

To overcome the aforementioned limitations, this research project explores the implementation of advanced computational methods to develop a generic, flexible and robust automated detection method for archaeological objects in remotely sensed data. More specifically, this project aims to develop user-friendly workflows for the detection of multiple classes of archaeological objects in LiDAR (Light Detection And Ranging; Wehr & Lohr 1999) data using Deep Learning (Goodfellow, Bengio & Courville 2016). The research project, a four-year PhD, is part of the Data Science Research Programme (DSRP) at the Faculty of Archaeology and the Leiden Centre of Data Science (LCDS) at Leiden University. The DSRP aims to bring together domain knowledge and associated 'big data' problems (for instance in archaeology) with the technical methods and solutions from data science.

This paper presents the results of the first year of the PhD project consisting of the first workflow developed, called WODAN (Workflow for Object Detection of Archaeology in the Netherlands). WODAN has successfully been implemented on LiDAR data from the research area in the Netherlands (**Figure 1**). The workflow serves as a proof of concept, to demonstrate that by implementing deep learning techniques it is possible to create a multi-class

Leiden University, NL

Corresponding author: Wouter Baernd Verschoof-van der Vaart (w.b.verschoof@arch.leidenuniv.nl)

detector for archaeological objects. While the first results are promising, there is still improvement needed in order to achieve a generic detection method.

In the following the research area is introduced (Section 2), followed by an overview of the type of deep learning technique used (Section 3). The structure of the workflow and the datasets (Section 4) as well as the results of the initial experiments (Section 5) will be presented and discussed (Section 6). The paper will finish with an overview of improvements and future developments planned for WODAN (Section 7).

2 The Research Area

The research area comprises a largely forested area of circa 2350 km² (about 7% of the total area of the Netherlands, excluding water) in the central part of the Netherlands (**Figure 1**). It is locally known as the *Utrechtse Heuvelrug* (western part) and the *Veluwe* (eastern part), which are separated by the *Gelderse Vallei*. Both the Utrechtse Heuvelrug and the Veluwe consist of ice-pushed ridges formed in the Saale glacial period (circa 350,000 to 130,000 years ago), and were subsequently covered with coversand deposits during the Weichselian glacial period (circa 115,000 to 10,000 years ago; Berendsen 2004). Till the second half of the Middle Ages (500 to 1500 AD),

the area remained largely covered by forest and heath. Between the 8th and 10th century AD large swathes of forest were cut down due to extending agricultural areas and iron production, the latter needing large amounts of charcoal. Areas with drift-sand (Aeolian sand) emerged, presumably due to the deforestation (Berendsen 2004). In the first quarter of the 20th century large parts of the research area were reforested and the majority of the still extant archaeological objects are now under forest cover. While this has likely contributed to their present day preservation, their location also hinders the survey of these archaeological objects (see also Kenzler & Lambers 2015). Nevertheless, the area holds one of the largest clusters of known (extant) archaeological objects in the Netherlands, including barrows, Celtic fields, charcoal kilns, hollow roads, and *landweren* (border barriers).² The area has been the subject of several (recent) archaeological projects (e.g. Arnoldussen 2018; Bourgeois 2013), providing up-to-date inventories of archaeological objects and their state of preservation (though see Section 4.1). This area is ideally suited for the research at hand thanks to the large number and overall good preservation of archaeological objects, their clear visibility in LiDAR data, the available archaeological information as well as the size of the area.

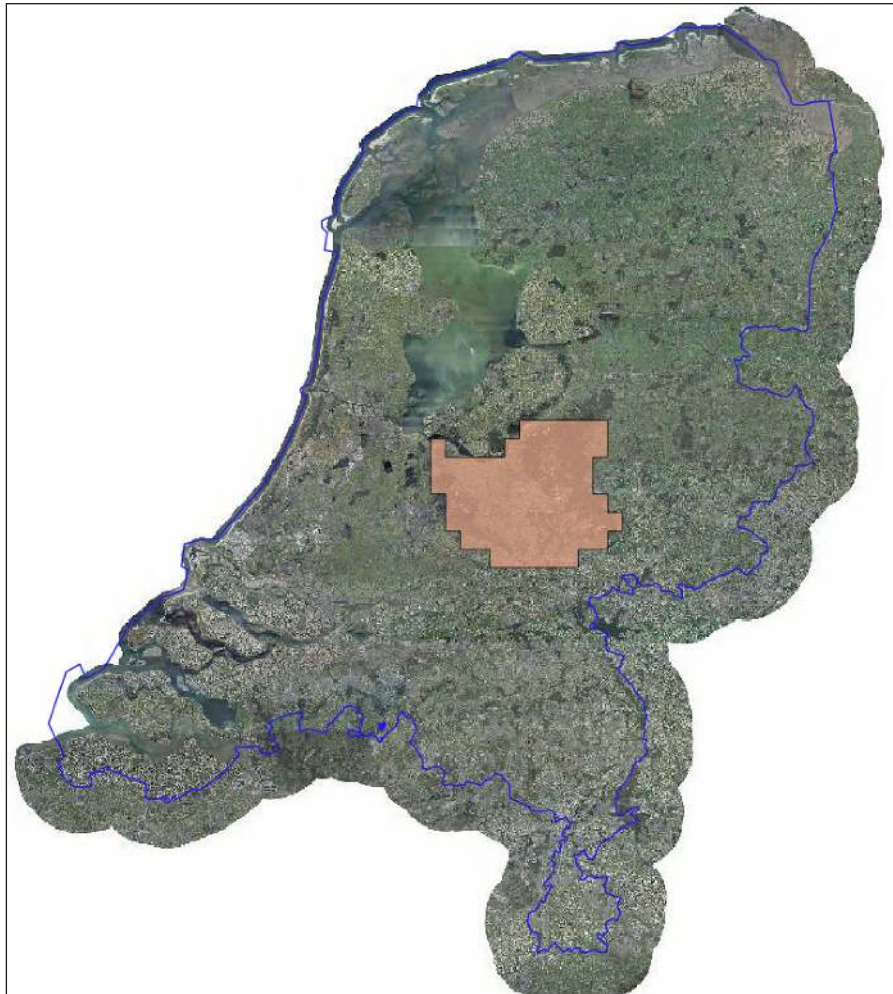


Figure 1: The research area (highlighted red) in the central part of the Netherlands (source background map <https://www.pdok.nl>).

3 From CNN to Faster R-CNN

Deep learning (DL) is a subfield of machine learning which attempts to acquire high-level abstractions in data by utilising hierarchical architectures (Guo et al. 2016: 27). Recently, DL has been applied in multiple artificial intelligence domains, such as computer vision and natural language processing. To date, the most frequently used DL architectures are Convolutional Neural Networks (CNNs; Krizhevsky, Sutskever & Hinton 2012). A CNN is an image-classifying algorithm that is loosely inspired by the human visual cortex (Ball, Anderson & Seng Chan 2017: 5). A typical CNN consists of an input layer, multiple hidden layers, and an output layer. The hidden layers are generally a combination of alternating convolutional- and pooling layers, followed by several fully-connected layers. In the convolutional layers, various filters (kernels) are used to convolve (add values of a pixel within an image to its neighbouring pixels based on a certain filter) the image into feature maps. The subsequent pooling layer reduces the dimensions of these feature maps (**Figure 2**).

After the last pooling layer, there are several fully-connected layers that look at to which particular class the produced feature maps most strongly correlate and compute the probabilities for the different classes (Guo et al. 2016: 28–29). Together all layers comprise a feature extractor and a classifier, of which the latter assigns class labels or computes probabilities of a given class being present in the input image (Ball, Anderson & Seng Chan 2017: 5).

A CNN learns from given examples (generally a very large set of labelled images), rather than relying on a human programmer to formulate rules or set parameters. The training of a CNN involves the following steps: forward-propagation, computation of the loss cost (error), and back-propagation. During forward-propagation the input image is fed through the different layers with the current parameters (weights and bias) fixed. The output is compared to the ground truth labels (the same manually labelled image) and used to calculate the loss cost. Based on the loss cost, the gradients of each parameter

are computed and used to update all parameters during back-propagation. All layers are then prepared for the next forward-propagation. One training round (a forward-propagation, computation of the loss cost (error), and back-propagation) of all training examples is called an epoch. After a sufficient number of these epochs—when the loss cost has become acceptably low—the training of the CNN can be stopped (Guo et al. 2016).

One of the main problems to overcome in (semi-)automated archaeological object detection is the (in comparison to other fields) small datasets that are available. One of the merits of CNNs (especially for a field such as archaeology) is the possibility of *transfer learning* or *domain adaptation*. Rather than training a CNN from scratch on a small dataset, a CNN is pre-trained on a generic image set and then optimised and reused on a small dataset (from a different domain; Razavian et al. 2014). Because all layers (except for the output layer) can use the pre-trained parameters during transfer learning, the training time of the CNN is greatly reduced while the generalisation ability of the algorithm is improved (Guo et al. 2016: 30). Transfer learning has been successfully applied in archaeological contexts, on photographs and drawings (Hohl 2016) as well as on images from remote sensing surveys (Trier, Cowley & Waldeland 2018; Trier, Salberg & Pilo 2018; Zingman et al. 2016).

However, in archaeological prospection the focus lies not only on characterising objects (or classifying, the typical task of a CNN) but also on obtaining the exact position (or localising) of these objects in the wider landscape (David 2005). This is where R-CNNs (Regions with CNN features or Region-based CNNs; Girshick et al. 2014) can prove useful. Unlike normal CNNs, which classify the entire input image, R-CNNs address the problem of object detection, which requires correctly localising and classifying (multiple) objects within a larger image (Guo et al. 2016: 39). The basic concept of an R-CNN is to generate multiple object proposals within an image, extract features from each proposal using a CNN, and then classify these. The developed

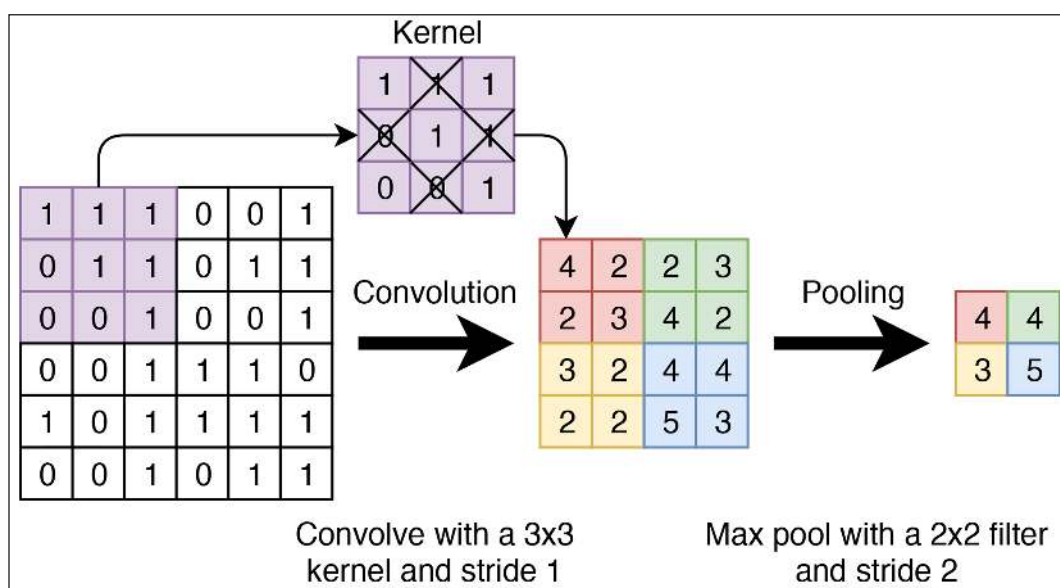


Figure 2: Schematic representation of a convolution- and pooling layer in a CNN.

workflow uses a recent ‘evolution’ of R-CNN: Faster R-CNN (Ren et al. 2017). For a detailed explanation of the workings of Faster R-CNN see Section 4.2.

4 WODAN

The aim of the research project is to create user-friendly workflows for multi-class archaeological object detection in remotely sensed data. The first workflow developed, called WODAN, is designed to detect barrows, Celtic fields and charcoal kilns in LiDAR data by utilising Faster R-CNN (Figure 3). WODAN can be broken down into three separate parts: a preprocessing- (Section 4.1), an object detection- (Section 4.2), and a post-processing part (Section 4.3). The first part deals with all the (pre)processing necessary to convert LiDAR data into input images that meet the requirements of Faster R-CNN. The second part of the workflow is where the actual object detection by Faster R-CNN is performed. In the post-processing part, the results of the object detection are converted back into geographical data. The latter is done to make the results more usable for archaeological prospection. In the following the different parts of the workflow are discussed.

4.1 Datasets

In order to successfully train Faster R-CNN for the task at hand, a training-, validation-,³ and testing dataset of LiDAR images containing labelled archaeological objects is needed. Unfortunately, at the outset of the project no such datasets were available, and therefore had to be created. The datasets were designed to be similar in format (image size, image type, and folder structure) to the PASCAL Visual Object Challenge (VOC) datasets (Everingham et al. 2010). These challenging datasets are the most widely employed for the evaluation of object detection architectures and therefore are readily usable in most object detection algorithms (Guo et al. 2016).

From the Veluwe (see Section 2) about 440 km² of interpolated LiDAR data was acquired.⁴ The interpolated data was visualized with Simple Local Relief Model (SLRM) from the *Relief Visualisation Toolbox* (Kokalj 2013). This visualization enhances the local detail, while suppressing the large-scale terrain (Hesse 2010). The images were subsequently turned into a format that Faster R-CNN requires (JPG files of maximum 1000 by 600 pixels) by importing them into *QGIS* (2.18 Las Palmas; QGIS Development Team

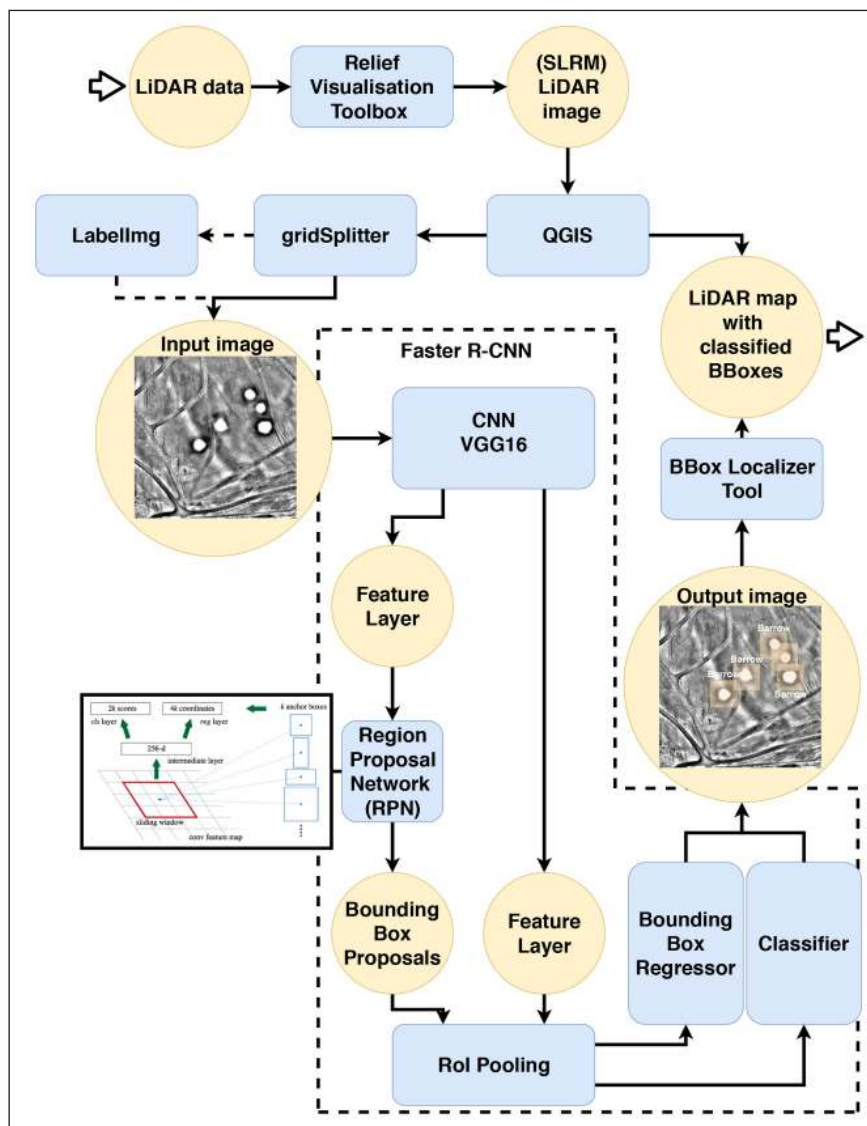


Figure 3: The workflow WODAN, with processes in blue and in-/output files in yellow.

2017) and using the plugin *gridSplitter* (Krambach 2016) (see also **Figure 3**). This resulted in 2940 sub-images of 1000 by 596 pixels (500 by 298 m). Due to this ‘cutting up’ of the images, about 3% of the barrows in the training dataset were dissected. It is unclear whether this had any effect on the training or performance of the model. However, it has been noted that there are also some cases where barrows have been dissected ‘naturally’ by other, more modern objects, such as (hollow) roads. This potential problem will be addressed in future research by implementing overlap between sub-images. The sub-images were converted from GeoTIFF files to JPG files with *Irfanview* (Skiljan 2005). All barrows, Celtic fields and charcoal kilns within these sub-images were labelled using *Labellmg* (Tzutalin 2015). However, during the labelling a problem arose. Many of the sub-images contained potential barrows that were previously unknown. At the moment of writing 745 potential barrows have been discovered in the datasets. As it was unclear whether these objects are real barrows or other (geological) objects, sub-images containing only potential barrows were excluded from the datasets. Potential barrows in sub-images containing both known and unknown objects were labelled as barrows. It was also noted that parts of the research area were made up of large zones of drift-sand of unknown date (but see

Bourgeois 2013). Sub-images containing large parts of this drift-sand were excluded as well, because the small dunes within these areas are indistinguishable from barrows, even for humans. In total 754 sub-images were excluded from the training- and validation datasets, based on the above arguments. A further 1360 sub-images contained no relevant archaeological objects and were excluded from the training- and validation datasets as well. For the testing dataset 73 sub-images (of the 420) were selected, including both sub-images with and without relevant archaeological objects. Several sub-images, containing ‘difficult’ terrain objects that could easily be mistaken for archaeological objects (objects of confusion), were added as well (see **Figure 4**). This resulted in training-, validation-, and testing datasets of respectively 365, 41, and 73 sub-images. See **Table 1** for more information on the datasets.

4.2 Faster R-CNN

The main part of the workflow consists of an altered version of Faster R-CNN.⁵ This particular R-CNN architecture was chosen because it performs well on several difficult object detection tasks (Guo et al. 2016: 39). The basic concept of an R-CNN is to generate multiple object proposals within an image, extract features from each proposal

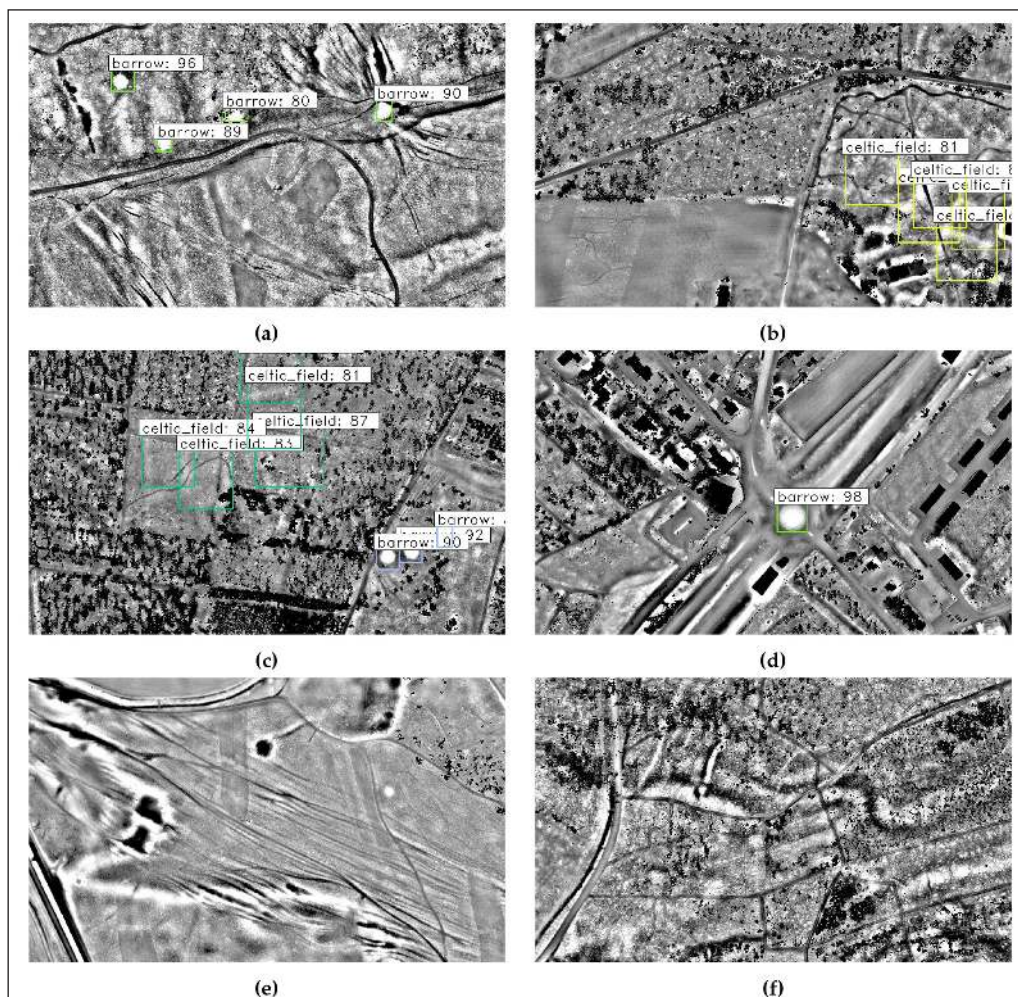


Figure 4: Output images of Faster R-CNN showing correctly detected barrows **(a)**; correctly detected Celtic fields **(b)**; both classes correctly detected within a single image **(c)**; a false detection of a roundabout classified as barrow **(d)**; undetected charcoal kilns (center left) **(e)**; ‘empty’ image with possible objects of confusion **(f)**.

Table 1: The developed datasets used in this research.

dataset	# images	# barrows	# Celtic fields	# char-coal kilns	# objects
training	365	749	904	119	1772
validation	41	49	199	24	272
testing	73	78	235	23	336

Table 2: Initial results of the experiments (values before slash are for barrows, after slash for Celtic fields).

Experiment	# epochs	Anchorbox sizes	Recall	Precision	F1	MaF1
1	12	16, 64, 512	0.76/0.19	0.57/ 0.71	0.65/0.30	0.43
2	15	16, 64, 128	0.78/0.48	0.36/0.47	0.49/0.47	0.47
3	15	16, 64, 256	0.69/ 0.97	0.77/0.26	0.73/0.41	0.45
4	15	16, 64, 384	0.71/0.92	0.90 /0.26	0.79 /0.41	0.46
5	15	16, 64, 512	0.62/0.82	0.55/0.58	0.59/ 0.68	0.66
6	18	16, 64, 512	0.81 /0.20	0.68/0.50	0.74/0.29	0.44

using a CNN, and then classify these. In the Faster R-CNN architecture (also see **Figure 3**) a region proposal network (RPN; a small fully convolutional neural network) generates the object proposals. The feature extraction and classification is done by the Fast R-CNN detector (Girshick 2015). Fast R-CNN consists of a bounding box regressor and a CNN (in this research the Resnet50- (He et al. 2016) or the VGG16 model (Simonyan & Zisserman 2015) was used). Both the RPN and the Fast R-CNN detector are trained simultaneously during the training of Faster R-CNN. The RPN takes an image as input and outputs a set of rectangular object proposals, each with a likelihood that the proposal contains a relevant object. The RPN is slid over the image with a set interval (called stride). At every location, multiple object proposals (or regions of interest; RoIs) are generated based on anchors of three different scales (also see **Table 2**) and three aspect ratios (1:1, 1:2, and 2:1) resulting in nine anchor boxes per location. These anchor boxes are fed into a regression layer, which outputs the coordinates of the anchor boxes, and into a classification layer that estimates the probability of the anchor box containing an object or no object. Simply said: the RPN tells the Fast R-CNN detector where to look (Ren et al. 2017: 3–4). The Fast R-CNN detector takes the image and the object proposals (that presumably contain an object) from the RPN. For every object proposal a probability of it belonging to a particular class (plus a catch-all ‘background’ class) is given (by the classifier) as well as a set of refined bounding box coordinates (by the bounding box regressor; Girshick 2015). Based on a set probability threshold object proposals are discarded or given as output of the model.

Several alterations were made to the ‘original’ Faster R-CNN model (see also **Table 2**):

- A validation step was incorporated after every two epochs to monitor overfitting of the model during training. During this step the model’s performance

was tested against the validation dataset. Decline in the performance during the validation step can be an indication of overfitting;

- Data augmentation was implemented by using horizontal- and vertical flip and 90° rotations (effectively multiplying the dataset by 4) to reduce overfitting;
- Because a pre-trained model was used on a small dataset the number of epochs was drastically lowered from 2000 to 12–18;
- The size of the anchor boxes was adjusted from 64, 128, 512 pixels to 16, 64, 128/256/384/512 pixels to better cope with the small objects in the images.

4.3 Post-processing

The prior object detection step results in the input images with superimposed bounding boxes (**Figure 4**). Every bounding box is associated with a class label and a confidence score (range 0–100). Furthermore, a text-file with class labels, confidence scores, and bounding box coordinates per image is created. One of the drawbacks of Faster R-CNN is that these bounding box coordinates are in a local coordinate system—instead of a ‘real-world’ coordinate system—based on the pixels of the input image. To convert the bounding box coordinates into coordinates usable in a GIS environment the *BoundingBox Localizer Tool* (BLT) was developed. This program takes the ‘real-world’ coordinates from the input images (while still in GeoTIFF format; see Section 4.1) and uses these to calculate the corner and central coordinates of the bounding boxes. The resulting points are used to create bounding box polygons with the convex hull algorithm in *QGIS* (QGIS Development Team 2017). The outcome is a GIS layer with the results of the object detection step (bounding boxes, class labels, and confidence scores) in ‘real-world’ coordinates. The results can therefore directly be verified with the original LiDAR data (or additional data) in a GIS environment.

5 Initial Results

The developed datasets (see Section 4.1) were used to train and test the altered Faster R-CNN model. Training and testing was done on one Graphics Processing Unit (GPU; NVidia Tesla K80). Training took between 20–30 minutes per epoch, while validation only took circa 2 minutes. On average the testing time was less than one second per image. A total of 20 experiments were conducted. Several parameters were varied between experiments in order to investigate their influence on the performance of the model – namely the ‘backbone’ CNN, the number of epochs, the stride of the RPN, and the size of the anchor boxes.

Nine experiments were conducted with Resnet50 (He et al. 2016) instead of VGG16 as the ‘backbone’ CNN. Both models (Resnet50 and VGG16) are designed to classify multiple classes. However, in the nine experiments conducted in this research with the Resnet50 model, Resnet50 was only able to detect a single class (barrows), not multiple classes (barrows and Celtic fields). While multiple Resnet50 models, one for each class, could be used in conjunction (see for instance Trier, Cowley & Waldeland 2018 for a comparable approach), this is not preferable as the research aims at a single, multi-class detector for various archaeological object classes. Further experiments with Resnet50 were dropped in favour of experiments with the VGG16 model. Another five experiments yielded no results: they were unable to detect any objects at all. This was caused either by overfitting or by changes in the model that resulted in critical failures, such as the changing of the stride of the RPN. **Table 2** shows the results of the six experiments that were able to detect multiple classes. In these experiments the number of epochs was varied between 12 and 18. The size of the anchor boxes was varied between 16, 64, 128 and 16, 64, 512.

The performance of the experiments was evaluated by calculating the *recall*, *precision*, *F1-score*, and *MaF1-score* (see **Table 2**). All these metrics reach their best value at 1 and worst at 0. These metrics are calculated by determining the number of true positives (TP), false positives (FP), and false negatives (FN). The recall gives a measure of how many relevant objects are selected ($\text{recall} = \text{TP}/(\text{TP}+\text{FN})$; Sammut & Webb 2010, 781). The precision measures how many of the selected items are relevant ($\text{precision} = \text{TP}/(\text{TP}+\text{FP})$; Sammut & Webb 2010, 781). The F1-score is the harmonic average of the precision and recall and a measure of the model’s performance per class ($\text{F1} = 2 \cdot \text{recall} \cdot \text{precision}/(\text{recall}+\text{precision})$; Sammut & Webb 2010, 397). The latter is usually more useful than accuracy, especially if the datasets have an uneven class distribution. The Micro averaged F1-score (MaF1) is the harmonic mean of the Micro averaged precision ($=\text{TP1}+\text{TP2}/(\text{TP1}+\text{TP2}+\text{FP1}+\text{FP2})$) and Micro averaged recall ($=\text{TP1}+\text{TP2}/(\text{TP1}+\text{TP2}+\text{FN1}+\text{FN2})$) and gives a measure of the model’s overall performance (Manning, Raghavan, & Schütze 2009, 280).

The results of the experiments (see **Table 2**) show a top performance of the Faster R-CNN model of 0.66 (MaF1) and an on average performance of 0.49 (average MaF1 score of all experiments). The experiments show that Faster R-CNN can be trained to detect and categorise both barrows and

Celtic fields (**Figure 4**). However, it should be noted that during all experiments Faster R-CNN was unable to detect charcoal kilns (see below). Charcoal kilns were therefore omitted from the results. The performance of Faster R-CNN varies greatly between experiments. Recall values for barrows and Celtic fields are respectively 0.62–0.81 (on average 0.73) and 0.19–0.97 (on average 0.60). Precision values for barrows and Celtic fields are respectively 0.36–0.90 (on average 0.64) and 0.26–0.71 (on average 0.46). F1-scores lie between 0.49 and 0.79 for barrows (on average 0.67) and between 0.29 and 0.68 for Celtic fields (on average 0.43). The latter shows that the model performs better at detecting barrows than Celtic fields.

6 Discussion

While Faster R-CNN performs adequately, with a top performance of 0.66 (MaF1) and an average MaF1-score of 0.49 (see **Table 2**), the method is still far from perfect. As stated above, Faster R-CNN has been unable to detect charcoal kilns in any of the conducted experiments (see **Figure 4**). Other studies have managed to (semi-)automatically detect charcoal kilns in LiDAR data (Schneider et al. 2015; Trier, Salberg & Pilø 2018). Performance issues in these studies were either contributed to the size (Schneider et al. 2015) and ‘high speciality’ (or complexity) of charcoal kilns (Trier, Salberg & Pilø 2018), the terrain (and terrain objects) surrounding the charcoal kilns (e.g. forest roads intersecting charcoal kilns; Schneider et al. 2015), or the difference between RGB photographs and greyscale LiDAR images (Trier, Salberg & Pilø 2018). The above-mentioned issues seem not to have been the main problem in our experiments. We were able to detect other archaeological objects, of comparable size and complexity, which were on occasion intersected by (modern) terrain objects in greyscale LiDAR images. The most probable cause for the lack of learning of this particular class is the low number of examples in the training set (119 versus 749 and 904 for the other classes; see **Table 1**).

Comparing the results to other archaeological (semi-) automated detection methods proves difficult. The main complications are insufficient details on the number of true- and false positives (e.g. Cerrillo-Cuenca 2017) and/or the fact that additional circumstances have been of influence on the results. As an example of the latter: in the research of Kramer (2015) on the detection of barrows in LiDAR data, the dataset used for verification included an unknown amount of levelled barrows which were impossible to detect in LiDAR data and therefore increased the number of false negatives. Such complications make it either challenging or even impossible to calculate the above-mentioned metrics, or make us question the validity of the calculated metrics. However, in **Table 3** a rough comparison between the results of our best performing model (listed as WODAN) and other detection methods for barrows (or equivalent mounds) is depicted. The object detection method used is also shown (see Cheng & Han 2016 for an overview of these different methods). The results shown are the best performing detection method in that particular paper. The rough comparison shows that our approach is among the top in performance compared

Table 3: Overview of the results (recall, precision, F1-scores) of other research on the detection of barrows or equivalent mounds (TP: True Positives; FP: False Positives; FN: False Negatives).

Reference	Method	# TP	# FP	# FN	Recall	Precision	F1
Caspari et al. (2014)	Machine Learning	32	11	118	0.21	0.74	0.33
Kramer (2015)	GEOBIA	50	196	163	0.23	0.20	0.21
Trier, Zortea & Tønning (2015)	Template Matching	108	810	109	0.50	0.12	0.19
Freeland et al. (2016)	GEOBIA	168	34	68	0.71	0.83	0.77
Freeland et al. (2016)	Rules-based (<i>iMound</i>)	200	42	36	0.85	0.83	0.84
Sevara et al. (2016)	GEOBIA	1581	220	275	0.85	0.87	0.86
Cerrillo-Cuenca (2017)	GEOBIA	28	8000	15	0.61	0.002	0.003
Guyot, Hubert-Moy & Lorho (2018)	Machine Learning	2952	41	46	0.98	0.99	0.98
WODAN	Machine Learning (R-CNN)	55	6	23	0.71	0.90	0.79

to other methods. These results therefore show that Faster R-CNN is a promising technique for the detection of multiple classes of archaeological objects in LiDAR data.

7 Conclusions

This paper presents the results of the application of a promising new technique for the automated detection of archaeological objects in LiDAR data, based on R-CNN (Region-based CNN). This deep learning architecture has successfully been integrated into a workflow, called WODAN. This workflow incorporates the preprocessing of LiDAR data (into the required format), multi-class object detection with Faster R-CNN, and the conversion of the results of the object detection step into geographical data. An altered Faster R-CNN model has been trained (through transfer learning) and tested on LiDAR data gathered from the central part of the Netherlands. The results of the experiments (see **Table 2**) show that while Faster R-CNN performs adequately (the MaF1-score of all experiments is on average 0.49), the method still requires improvement. The model is able to detect and categorise two types of archaeological objects (barrows and Celtic fields), but has been unable to detect a third type (charcoal kilns), probably due to the low number of examples in the training dataset. Based on a rough comparison with other detection methods for barrows (or equivalent mounds), Faster R-CNN is among the top in performance (see **Table 3**). These results show that Faster R-CNN is a promising technique for the detection of multiple classes of archaeological objects in LiDAR data.

The research discussed in this paper is the results of the first year of a four-year PhD project and serves as a proof of concept for the usability of R-CNN architectures for archaeological object detection. In subsequent research, Faster R-CNN will be improved further by increasing the number of examples and classes in the training dataset, by addressing the potential problem of dissected objects within the datasets by implementing overlap between sub-images, by implementing additional data augmentation,

and by using models that have been pre-trained on data more comparable to LiDAR data. A method will be developed either to insert additional domain information (for instance about the subsoil, current land use etc.) into the LiDAR images or to use domain information in an additional classification step after the object detection. This will make the method useful in large-scale archaeological mapping over different types of terrain. WODAN will be improved by automating the steps within the workflow and incorporating these within a single program (or *QGIS* plugin). The end result will be a user-friendly application for multi-class archaeological object detection in remotely sensed data.

Notes

- ¹ In the field of Computer Vision the term ‘feature’ refers to the properties of an image, while an ‘object’ refers to real-world entities (Traviglia, Cowley & Lambers 2016: 14). Within this article the term ‘objects’ is therefore used for archaeological features, such as barrows.
- ² For instance, the Veluwe has one of the densest concentrations of barrows in the Low Countries, with more than 1000 recorded examples (Bourgeois 2013: 3).
- ³ A validation dataset was used in the experiments to monitor *overfitting* during training; a problem where a CNN has memorized the training examples, but it has not learned to generalise to new situations.
- ⁴ Classified, as well as interpolated LiDAR data of the entire Netherlands (with a point density of 6–10 per m² and a 50 cm resolution) is freely available from the *Actueel Hoogtebestand Nederland* (<https://ahn.arcgis-online.nl/ahnviewer/>).
- ⁵ A Keras (Chollet 2015) implementation of Faster R-CNN (<https://github.com/yhenon/keras-frcnn>) was used.

Competing Interests

The authors have no competing interests to declare.

References

- Arnoldussen, S.** 2018. The fields that outlived the Celts: The use-histories of later prehistoric field systems (Celtic fields or Raatakkers) in the Netherlands. *Proceedings of the Prehistoric Society*, 84: 303–327. DOI: <https://doi.org/10.1017/ppr.2018.5>
- Ball, JE, Anderson, DT and Seng Chan, C.** 2017. Comprehensive survey of deep learning in remote sensing: Theories, tools, and challenges for the community. *Journal of Applied Remote Sensing*, 11(4): 1–54. DOI: <https://doi.org/10.1117/1.JRS.11.042609>
- Bennett, R, Cowley, D and De Laet, V.** 2014. The data explosion: Tackling the taboo of automatic feature recognition in airborne survey data. *Antiquity*, 88(341): 896–905. DOI: <https://doi.org/10.1017/S0003598X00050766>
- Berendsen, H.** 2004. *De vorming van het land. Inleiding in de geologie en de geomorfologie*. Assen: Koninklijke Van Gorcum.
- Bourgeois, QPJ.** 2013. *Monuments on the Horizon. The formation of the barrow landscape throughout the 3rd and 2nd millennium BC*. Leiden: Sidestone Press.
- Caspari, G, Balz, T, Gang, L, Wang, X and Liao, M.** 2014. Application of Hough forests for the detection of grave mounds in high-resolution satellite imagery. In: *2014 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 906–909. Québec City, QC: IEEE. DOI: <https://doi.org/10.1109/IGARSS.2014.6946572>
- Cerrillo-Cuenca, E.** 2017. An approach to the automatic surveying of prehistoric barrows through LiDAR. *Quaternary International*, 435(B): 135–145. DOI: <https://doi.org/10.1016/j.quaint.2015.12.099>
- Cheng, G and Han, J.** 2016. A survey on object detection in optical remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 117: 11–28. DOI: <https://doi.org/10.1016/j.isprsjprs.2016.03.014>
- Chollet, F.** 2015. *Keras*. Available at: <https://keras.io> [Last accessed 18 July 2018].
- David, A.** 2005. The role and practice of archaeological prospection. In: Brothwell, D and Pollard, A (eds.), *Handbook of Archaeological Sciences*, 521–527. Chichester: John Wiley & Sons LTD.
- De Boer, A.** 2007. Using pattern recognition to search LIDAR data for archaeological sites. In: Figueiredo, A and Velho, G (eds.), *The World is in your Eyes: Proceedings of the XXXIII Computer Applications and Quantitative Methods in Archaeology Conference (March 2005 – Tomar, Portugal)*, 245–254. Tomar: CAA Portugal. DOI: <https://doi.org/10.15496/publikation-2797>
- De Laet, V, Paulissen, E and Waelkens, M.** 2007. Methods for the extraction of archaeological features from very high-resolution Ikonos-2 remote sensing imagery, Hisar (southwest Turkey). *Journal of Archaeological Science*, 34(5): 830–841. DOI: <https://doi.org/10.1016/j.jas.2006.09.013>
- Everingham, M, Van Gool, LI, Williams, CK, Winn, J and Zisserman, A.** 2010. The PASCAL Visual Object Classes (VOC) challenge. *International Journal of Computer Vision*, 88: 303–338. DOI: <https://doi.org/10.1007/s11263-009-0275-4>
- Freeland, T, Heung, B, Burley, DV, Clark, G and Knudby, A.** 2016. Automated feature extraction for prospection and analysis of monumental earthworks from aerial LiDAR in the Kingdom of Tonga. *Journal of Archaeological Science*, 69: 64–74. DOI: <https://doi.org/10.1016/j.jas.2016.04.011>
- Girshick, R.** 2015. *Fast R-CNN*, 27 September 2015. Available at: <https://arxiv.org/pdf/1504.08083.pdf> [Last accessed 18 July 2018].
- Girshick, R, Donahue, J, Darrell, T and Malik, J.** 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 580–587. Columbus, OH: IEEE. DOI: <https://doi.org/10.1109/CVPR.2014.81>
- Goodfellow, I, Bengio, Y and Courville, A.** 2016. *Deep Learning*. Cambridge, MA: The MIT Press.
- Guo, Y, Liu, Y, Oerlemans, A, Lao, S, Wu, S and Lew, MS.** 2016. Deep learning for visual understanding: A review. *Neurocomputing*, 187: 27–48. DOI: <https://doi.org/10.1016/j.neucom.2015.09.116>
- Guyot, A, Hubert-Moy, L and Lorho, T.** 2018. Detecting Neolithic burial mounds from LiDAR-derived elevation data using a multi-scale approach and machine learning techniques. *Remote Sensing*, 10(2): 225: 1–19. DOI: <https://doi.org/10.3390/rs10020225>
- He, K, Zhang, X, Ren, S and Sun, J.** 2016. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. Las Vegas, NV: IEEE. DOI: <https://doi.org/10.1109/CVPR.2016.90>
- Hesse, R.** 2010. LiDAR-derived local relief models – A new tool for archaeological prospection. *Archaeological Prospection*, 17(2): 67–72. DOI: <https://doi.org/10.1002/arp.374>
- Hohl, S.** 2016. *Neural network based image classification in the context of archaeology*. Unpublished thesis (Master), HTW Berlin.
- Kenzler, H and Lambers, K.** 2015. Challenges and perspectives of woodland archaeology across Europe. In: Giligny, F, Djindjian, F, Costa, L, Moscati, P and Robert, S (eds.), *CAA2014 – 21st Century Archaeology. Concepts, methods and tools. Proceedings of the 42nd Annual Conference on Computer Applications and Quantitative Methods in Archaeology*, 73–80. Oxford: Archaeopress.
- Kokalj, Ž.** 2013. *Relief Visualization Toolbox (RVT)*. Available at: <https://iaps.zrc-sazu.si/en/rvt#v> [Last accessed 18 July 2018].

- Krambach, M.** 2016. *gridSplitter*. Available at: <https://plugins.qgis.org/plugins/gridSplitter/> [Last accessed 18 July 2018].
- Kramer, I.** 2015. *An archaeological reaction to the remote sensing data explosion*. Unpublished thesis (Master), University of Southampton.
- Krizhevsky, A, Sutskever, I and Hinton, GE.** 2012. ImageNet classification with deep convolutional neural networks. *Advances In Neural Information Processing Systems*, 25: 1106–1114.
- Manning, CD, Raghavan, P and Schütze, H.** 2009. *Introduction to Information Retrieval*. Cambridge: Cambridge University Press.
- QGIS Development Team.** 2017. *QGIS Geographic Information System*. Available at: <https://qgis.org> [Last accessed 18 July 2018].
- Razavian, AS, Azizpour, H, Sullivan, J and Carlsson, S.** 2014. CNN features off-the-shelf: An astounding baseline for recognition. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR)*, 806–813. Columbus, OH: IEEE. DOI: <https://doi.org/10.1109/CVPRW.2014.131>
- Ren, S, He, K, Girshick, R and Sun, J.** 2017. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6): 1137–1149. DOI: <https://doi.org/10.1109/TPAMI.2016.2577031>
- Sammut, C and Webb, GI.** 2010. *Encyclopaedia of Machine Learning*. Boston, MA: Springer. DOI: <https://doi.org/10.1007/978-0-387-30164-8>
- Schneider, A, Talka, M, Nicolay, A, Raab, A and Raab, T.** 2015. A template-matching approach combining morphometric variables for automated mapping of charcoal kiln sites. *Archaeological Prospection*, 22: 45–62. DOI: <https://doi.org/10.1002/arp.1497>
- Sevara, C, Pregesbauer, M, Doneus, M, Verhoeven, G and Trinks, I.** 2016. Pixel versus object – A comparison of strategies for the semi-automated mapping of archaeological features using airborne laser scanning data. *Journal of Archaeological Science: Reports*, 5: 485–498. DOI: <https://doi.org/10.1016/j.jasrep.2015.12.023>
- Simonyan, K and Zisserman, A.** 2015. *Very deep convolutional networks for large-scale image recognition*, 10 April 2015. Available at: <https://arxiv.org/pdf/1409.1556.pdf> [Last accessed 18 July 2018].
- Skiljan, I.** 2005. *Irfanview*. Available at: <https://www.irfanview.com/> [Last accessed 18 July 2018].
- Travaglia, A, Cowley, D and Lambers, K.** 2016. Finding common ground: Human and computer vision in archaeological prospection. *AARGnews*, 53: 11–24.
- Trier, ØD, Cowley, DC and Waldeland, AU.** 2018. Using deep neural networks on airborne laser scanning data: Results from a case study of semi-automatic mapping of archaeological topography on Arran, Scotland. *Archaeological Prospection*, 1–11. DOI: <https://doi.org/10.1002/arp.1731>
- Trier, ØD, Salberg, A-B and Pilø, LH.** 2018. Semi-automatic mapping of charcoal kilns from airborne laser scanning data using deep learning. In: Matsumoto, M and Uleberg, E (eds.), *CAA2016: Oceans of Data. Proceedings of the 44th Conference on Computer Applications and Quantitative Methods in Archaeology*, 219–231. Oxford: Archaeopress.
- Trier, ØD, Zorzea, M and Tønning, C.** 2015. Automatic detection of mound structures in airborne laser scanning data. *Journal of Archaeological Science: Reports*, 2: 69–79. DOI: <https://doi.org/10.1016/j.jasrep.2015.01.005>
- Tzutalin.** 2015. *Labellmg. Git code*. Available at: <https://github.com/tzutalin/labellmg> [Last accessed 18 July 2018].
- Wehr, A and Lohr, U.** 1999. Airborne laser scanning—an introduction and overview. *ISPRS Journal of Photogrammetry & Remote Sensing*, 54(2–3): 68–82. DOI: [https://doi.org/10.1016/S0924-2716\(99\)00011-8](https://doi.org/10.1016/S0924-2716(99)00011-8)
- Zingman, I, Saupe, D, Penatti, OAB and Lambers, K.** 2016. Detection of fragmented rectangular enclosures in very-high-resolution remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 54: 4580–4593. DOI: <https://doi.org/10.1109/TGRS.2016.2545919>

How to cite this article: Verschoof-van der Vaart, WB and Lambers, K. 2019. Learning to Look at LiDAR: The Use of R-CNN in the Automated Detection of Archaeological Objects in LiDAR Data from the Netherlands. *Journal of Computer Applications in Archaeology*, 2(1), pp. 31–40. DOI: <https://doi.org/10.5334/jcaa.32>

Submitted: 14 January 2019

Accepted: 20 February 2019

Published: 19 March 2019

Copyright: © 2019 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

]u[

Journal of Computer Applications in Archaeology, is a peer-reviewed open access journal published by Ubiquity Press.

OPEN ACCESS 