

Article

# Comparing Three Machine Learning Techniques for Building Extraction from a Digital Surface Model

Nicla Maria Notarangelo , Arianna Mazzariello, Raffaele Albano \*  and Aurelia Sole 

School of Engineering, University of Basilicata, Via dell'Ateneo Lucano 10, 85100 Potenza, Italy; nicla.notarangelo@gmail.com (N.M.N.); arianna.mazzariello@unibas.it (A.M.); aurelia.sole@unibas.it (A.S.)  
\* Correspondence: raffaele.albano@unibas.it

**Abstract:** Automatic building extraction from high-resolution remotely sensed data is a major area of interest for an extensive range of fields (e.g., urban planning, environmental risk management) but challenging due to urban morphology complexity. Among the different methods proposed, the approaches based on supervised machine learning (ML) achieve the best results. This paper aims to investigate building footprint extraction using only high-resolution raster digital surface model (DSM) data by comparing the performance of three different popular supervised ML models on a benchmark dataset. The first two methods rely on a histogram of oriented gradients (HOG) feature descriptor and a classical ML (support vector machine (SVM)) or a shallow neural network (extreme learning machine (ELM)) classifier, and the third model is a fully convolutional network (FCN) based on deep learning with transfer learning. Used data were obtained from the International Society for Photogrammetry and Remote Sensing (ISPRS) and cover the urban areas of Vaihingen an der Enz, Potsdam, and Toronto. The results indicated that performances of models based on shallow ML (feature extraction and classifier training) are affected by the urban context investigated (F1 scores from 0.49 to 0.81), whereas the FCN-based model proved to be the most robust and best-performing method for building extraction from a high-resolution raster DSM (F1 scores from 0.80 to 0.86).

**Keywords:** automated building extraction; machine learning (ML); deep learning (DL); digital surface model (DSM); histogram of oriented gradients (HOG); support vector machine (SVM); extreme learning machine (ELM); fully convolutional network (FCN)



**Citation:** Notarangelo, N.M.; Mazzariello, A.; Albano, R.; Sole, A. Comparing Three Machine Learning Techniques for Building Extraction from a Digital Surface Model. *Appl. Sci.* **2021**, *11*, 6072. <https://doi.org/10.3390/app11136072>

Academic Editor: Hyung-Sup Jung

Received: 30 May 2021  
Accepted: 28 June 2021  
Published: 30 June 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In recent years, the availability of high-spatial-resolution remote sensing data has fostered the development of research methods and applications in several fields [1], such as urban planning, land monitoring, and environmental risk management [2–5] (e.g., urbanization delineation, vegetation mapping, flood modeling). The accurate information provided by the high-resolution data has been exploited both in large-scale problems, such as land use and land cover types [6,7], and small-scale ones, such as the extraction of urban objects: trees, roads, buildings, etc. [8–10]. Laser imaging detection and ranging (LiDAR) airborne mapping systems supply point clouds datasets containing 3-dimensional x, y, and z points and attributes to produce precise digital terrain model (DTM) and digital surface model (DSM) products—within a gridded or raster data format—in both natural and manmade environments [11].

Automatic building extraction from remotely sensed data is a major area of interest for an extensive range of applications but challenging due to difficulties in extracting precise boundaries because of urban morphology complexity [12]. Different methods have been proposed to address this issue [13], such as methods based on template matching, knowledge, object-based image analysis, and machine learning (ML) [14]. Commonly used template-matching-based approaches for automated buildings and urban objects footprints detection use “snake” or active contour model (ACM) [15] improvements [10,16–18] and

integration with ML and deep learning (DL) [19]. The approaches based on supervised ML can achieve the best results, especially along with DL [14], a subset of ML based on neural networks with representation learning [20]. Traditional computer vision techniques require manually engineered feature descriptors for the desired object detection, whereas DL models automate the process of feature engineering [21,22]. Generally, classical ML-based object detection methods involve two steps: feature extraction and classifier training. Among features descriptors, histogram of oriented gradients (HOG) [23] features have proven to be effective in describing the edge or local shape information of the urban objects [14,24–27]. Typical ML algorithms for classifier training include, but are not limited to, the support vector machine (SVM) [7,28–30], artificial neural network (ANN) [31,32], including the extreme learning machine (ELM) [33] and AdaBoost. Several successful buildings and urban objects detection approaches are based on DL methods [10,34–37], such as convolutional networks, and in particular, fully convolutional networks (FCNs) have shown good performance on semantic segmentation [38–42], with correct pixel classification and accurate spatial information [43].

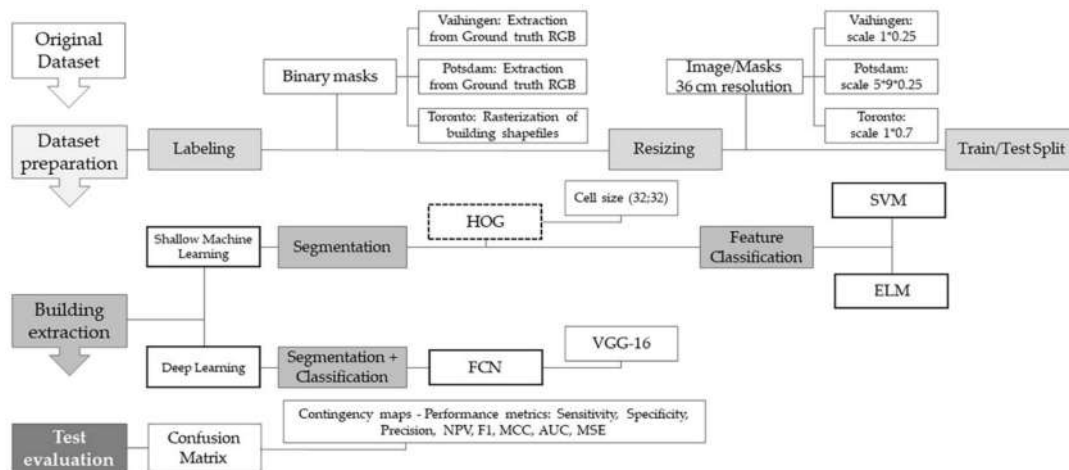
Despite the numerous algorithms proposed, building segmentation based only on the geometric information provided by DSM data is still a complicated task [44,45], mainly because objects with similar morphological characteristics and height can create ambiguity, resulting in position inaccuracy and local under-sampling [35,45].

A simplified method to detect and extract the building footprint based only on a DSM as input without any other additional feature could be beneficial in scenarios where only DSM data are available and a more expeditious solution is desirable (e.g., the rapid assessment of building damage). This paper aims to investigate the capability of three different, popular supervised ML models—namely HOG with SVM, HOG with ELM, and FCN—to detect building footprints using only raster DSM data as input and evaluate their performance on a publicly available benchmark dataset. This empirical comparison may highlight the potential usefulness of expeditious methods and help understand which model performs best in different urban environments. The first two methods rely on a HOG feature descriptor and a classical ML classifier (SVM) or a shallow neural network classifier (ELM) [46–48], and the FCN model is based on DL [49,50]. The descriptor and classification models were chosen for their accuracy [14,51], especially in segmentation tasks [52], popularity [14], simplicity, reduced computational time, and the ability to extract building footprint masks without reducing the resolution of raster data.

## 2. Materials and Methods

### 2.1. Overview

Figure 1 shows the general workflow of our study. The core steps included data retrieval and preparation, model implementation and training, and test performance evaluation. All the algorithms were implemented using the MATLAB environment, except the FCN that was implemented with the Python programming language and the Torch package. The experiments were conducted on a Fujitsu Celsius Workstation with Intel Xeon E5-2643 CPU (3.30 GHz) and 16 GB RAM. The following subsections describe each step.



**Figure 1.** Overview of the study workflow steps: data retrieval and preparation (labeling, resizing, and train/test split), model implementation and training (building extraction), and test performance evaluation.

## 2.2. Dataset Description

The data used in this study were obtained from the International Society for Photogrammetry and Remote Sensing (ISPRS) “Test Project on Urban Classification, 3-D Building Reconstruction, and Semantic Labeling”. The sets consisting of airborne images and laser scanner data were made publicly available to evaluate and compare different urban object extraction methods, providing benchmark datasets with ground truth [13,53], more updated and complete than the existing ones [54–56].

For the sake of generalizability, the available datasets were chosen covering three urban areas that differ in terms of urban morphology.

The first part of the dataset was captured over the city of Vaihingen an der Enz (Germany) and originally obtained from the digital aerial cameras tests carried out by the German Association of Photogrammetry and Remote Sensing (Deutsche Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation, DGPF) [57]. Each of the 33 different-sized patches (Figure 2) of the Vaihingen dataset contained a labeled true orthophoto (TOP) paired with a DSM [58] with a ground sampling distance of 9 cm. The publishers of the dataset provided a train/test split (30 patches for training, outlined with red, and 3 for testing, outlined with green). The morphology of Vaihingen presents the characteristics of a relatively small Central European village with many detached buildings and small multi-story buildings. It encompasses slightly different urban fabrics: relatively dense patterns formed by historic buildings with complex shapes and sparse trees, loose patterns formed by few high-rising residential buildings surrounded by trees, and regular patterns formed by small detached houses.

The second part of the dataset was captured over the city of Potsdam (Germany) and was originally obtained from the DGPF tests [58]. The Potsdam dataset consisted of 38 patches containing a labeled TOP paired with a DSM with a  $6000 \times 6000$  pixel size and a ground-sampling distance of 5 cm. The provided train/test split for the Potsdam dataset (Figure 3) consisted of 35 patches for training (outlined with red) and 3 for testing (outlined with green). The morphology of Potsdam represents a typical Central European historic city with large building blocks, narrow streets, and a dense settlement structure.

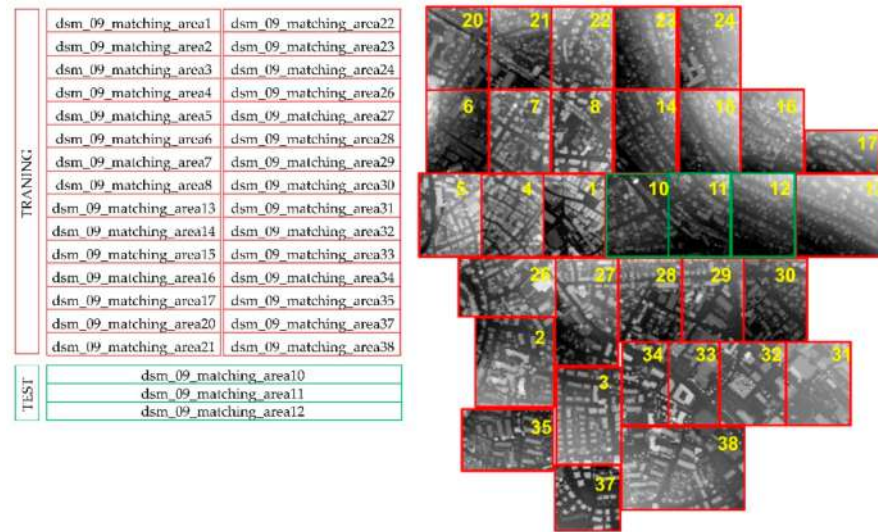


Figure 2. Digital surface model (DSM) patches of the Vaihingen an der Enz dataset: training (red) and test (green) subsets.

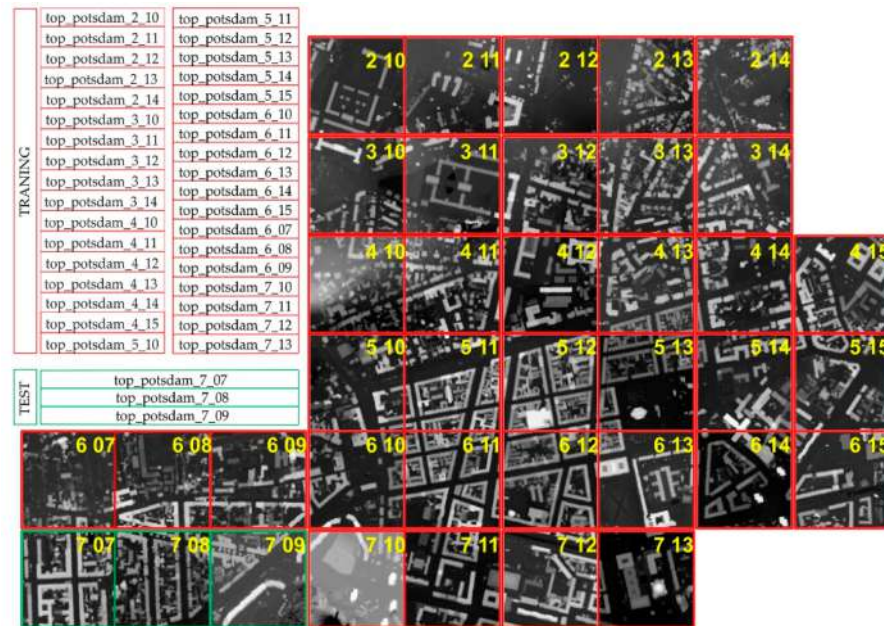
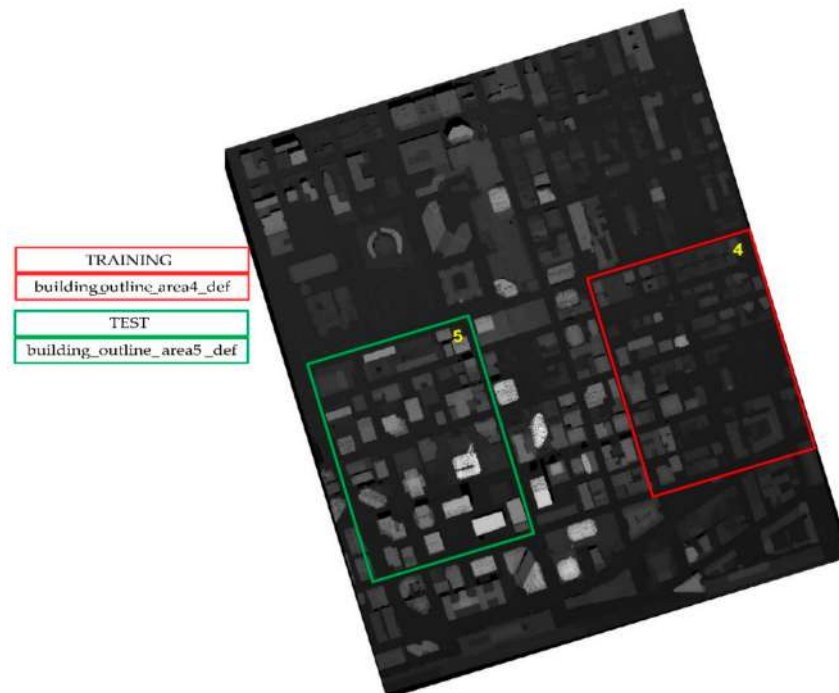


Figure 3. Digital surface model (DSM) patches of the Potsdam dataset: training (red) and test (green) subsets.

The third part of the dataset covered the city of Toronto (Canada) and consisted of 3 different-sized patches containing a TOP and a DSM interpolated from the airborne laser scanner point cloud with a grid width of 25 cm. The given train/test split ratio for the Toronto dataset (Figure 4) was 2:1. The urban morphology of downtown Toronto exhibits the characteristics of a typical modern North American megacity as the presence of different urban objects and urban fabrics formed by a mixture of low- and high-story buildings with various degrees of shape complexity in rooftop structures and streets, including clusters of high-rise buildings that cast fairly large shadows [59].

The final database retrieved from the different ISPRS challenges was representative of the typical Central European and North American cities with respect to urban complexity, dimensions, and fabric density, especially suitable for method evaluation and comparison.

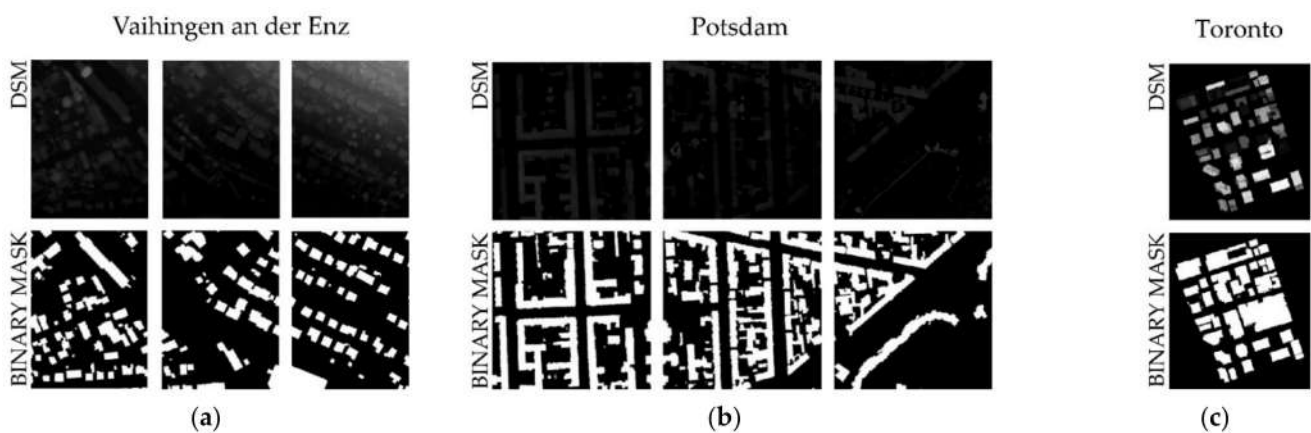




**Figure 4.** Digital surface model (DSM) patches of the Toronto dataset: training (red) and test (green) subsets.

The supervised learning models require a labeled set of data to learn, during the training process, the underlying patterns that can be used to make predictions on novel data.

To solve the building detection task, the building footprints were considered as a group of pixels that can be distinguished from the pixels representing any other objects that may appear in an urban environment (roads, paved areas, vegetation, etc.) through discriminatory features extracted from the DSM raster images. Thus, the input to the models and test data were lists of DSM tiles and the corresponding binary maps (Figure 5) containing the ground truth for two semantic classes: building and non-building. For the Vaihingen and Potsdam sets, the binary masks were generated by selecting the building category from the provided ground truth (8-bit RGB tif files with one color per land cover class). For the Toronto set, the 8-bit binary masks were obtained by rasterizing the shapefiles describing the building outlines.



**Figure 5.** Sample of the dataset of digital surface model (DSM) patches and corresponding binary masks (ground truth): (a) Vaihingen test set, (b) Potsdam test set, and (c) Toronto test set.

The preprocessing phase consisted of rescaling the raster data with an adequate factor and applying nearest interpolation (Figure 1). The initial mismatching resolutions—9 cm/pixel for the Vaihingen set, 5 cm/pixel for the Potsdam set, and 25 cm/pixel for the Toronto set—were conformed to a unique value of 36 cm/pixel to improve computational speed and performances. This value was obtained by progressively down-scaling a sample until a marked computational time reduction was seen without significant loss of information. The adjusted set of DSM tiles and masks was binned into training and test subsets.

Despite representing different and complex urban morphologies, the generated dataset showed a balanced class distribution. The number of pixels belonging to the building class and non-building class was close to the ideal 1:1 ratio (percentage difference  $\ll 0.5\%$ ) in every tile. Furthermore, the quality of the original datasets ensured the absence of uncertainty in labeling due to noise, low resolution, no-data pixels, inaccurate object edges, or class overlapping. Thus, the possible biases for the building extraction models were minimized, safeguarding the fairness of the test evaluation and the potential transferability of the learned knowledge.

### 2.3. Models' Implementation and Training

#### 2.3.1. Shallow ML-Based Building Detection

A shallow ML-based model that can classify pixels as building or non-building requires discriminative and computable engineered features for image segmentation and pixels classification.

Among the existing edge- and gradient-based descriptors, the HOG descriptors appeared especially suitable for detecting the construction footprints, as HOG detectors cue mainly on contours, can be computed quickly, and are fairly invariant to geometric transformations and occlusions [23].

The HOG feature extraction chain computes occurrences of gradient orientation in the detection window, or the region of interest (ROI) across the input, on a regular cell grid and uses overlapping local contrast normalization to enhance the accuracy. For every pixel in the input, the gradient vectors contain information on pixel value changes in  $x$  ( $g_x$ ) and  $y$  ( $g_y$ ) directions with respect to its four neighbors.

The attributes of the gradient are its magnitude

$$M(x, y) = \sqrt{g_x^2 + g_y^2} \quad (1)$$

and its direction

$$\theta = \arctan(g_y / g_x). \quad (2)$$

Gradient information is then pooled into a 1-D histogram of orientations that can be used as input for ML algorithms.

Thus, the feature extractor encoded the raster inputs into feature vectors to feed a classifier for the object/non-building instance detection, namely ELM and SVM classifiers.

The first method for building detection under examination used HOG as an input to the SVM [60], a supervised ML algorithm that produces accurate classifications of remotely sensed data [28,61–63]. The binary SVM classifier models a data point as a multi-dimensional vector belonging to one of two classes and constructs a hyperplane or set of hyperplanes to separate the classes [64] with the maximal margin (a space containing no observations) and the lowest misclassification [65–67].

The SVM classifier was fitted on the aforementioned processed training set to find the best-separating hyperplane (i.e., the decision boundary) using as input the predictor vectors  $X_j$  along with their class labels  $Y_i \in \{+1, -1\}$ . During the training process, sequential minimal optimization [68] solved the constrained optimization problem by breaking it into a series of sub-problems that could be solved analytically. The final model was then saved to perform prediction on the independent test set in the evaluation phase.

The second method combines the HOG and an ELM [47] classifier instead of the SVM [46]. The ELM trains a shallow feedforward neural network with a single hidden layer, i.e., the feature mapping, which does not need parameter tuning [69–71]. The main advantages of the ELM method compared to the previous one include better scalability and similar generalization at a faster learning speed [72].

The ELM classifier was fitted on the same training subset with a single hidden layer of 1000 neurons and a training ratio value of 0.9 to find the combination of nodes, weights, and biases minimizing the error between the actual output of the network (predictions) and the expected one (the ground truth) and obtain a learned model to be used for the test evaluation. The net started with random input weights and calculated the best values using the root-mean-square error to assess the prediction accuracy.

### 2.3.2. DL-Based Building Detection

The third building detection model was based on a DL architecture. The FCN-based semantic segmentation [50] classifies the building or non-building class for each pixel directly within the image inputs. The FCN gives a pixel-wise output (label map) without needing a hand-engineered feature vector to feed the classifier for building extraction of remotely sensed data [7,42,73–75].

The state-of-the-art classification convolutional model VGG-16 [76,77] structure was repurposed for the segmentation task employing the method fully described in [50]: adapting the original architecture into an FCN and transferring its learned representations by fine-tuning to the desired task. The repurposed architecture combines the semantic global information of deep, coarse layers to learn the feature hierarchy and the local knowledge of shallow, fine layers with 32-, 16-, and 8-pixel strides to improve the segmentation accuracy and enable pixel-wise predictions.

We fit the three models on a common portion of the dataset (the training set) by feature extraction and classifier training for the shallow ML-based models and fine-tuning through transfer learning for the DL model and then evaluated their discriminatory ability on the untouched data left (the test set).

### 2.4. Test Evaluation

The three implemented methods were fitted on a common dataset and then used to generate predictions on the same hold-out test set to obtain an unbiased estimate of each model's accuracy.

Given the building label as the positive class and non-building label as the negative class assigned to every single pixel of the raster images, the true positives (TPs), false positives (FPs), true negatives (TNs), and false negatives (FNs) were counted to obtain the pixel-wise metrics for the evaluation of the binary classifiers [78,79] and a contingency map from a pixel-to-pixel comparison [32,80,81].

The TPs were the pixels correctly identified as building pixels; the FPs were non-building pixels wrongly labeled as belonging to the building class. Similarly, the TNs were non-building pixels correctly classified, whereas the FNs were building pixels wrongly classified as non-building pixels (undetected building pixels). Totaling the number of TPs, TNs, FPs, and FNs added up to the total pixels of the test set.

Sensitivity, or recall, is the proportion of TPs, pixels that actually are positive (building class). Sensitivity takes values in the range (0, 1); with higher sensitivity, fewer building pixels are undetected (larger footprints).

$$\text{Sensitivity or Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (3)$$

Specificity is the proportion of TNs, pixels that actually are negative (non-building class). Precision takes values in the range (0, 1); higher specificity leads to fewer pixels mislabeled as building.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}, \quad (4)$$

The relationship between sensitivity and specificity was visualized using the receiver operating characteristic (ROC) curve and the area under the curve (AUC) [82] to quantify the performance of each classifier over its range of possible cut-offs (classification thresholds).

Precision or positive predicted value (PPV) is the proportion of relevant pixels (TP) among all the pixels classified as building pixels (positive) in the test. The PPV varies from 0 to 1, corresponding respectively to the value of the worst and the best classifier.

$$\text{Precision or PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (5)$$

Similarly, the negative prediction value (NPV) represents the proportion of pixels with accurate non-building test results (TN) among all the pixels identified as non-building pixels (negative class). NPV reference values are 0 for the worst classification and 1 for the best classification possible.

$$\text{NPV} = \frac{\text{TN}}{\text{TN} + \text{FN}}, \quad (6)$$

The F1 score combines the precision and recall values by taking their harmonic mean. The F1 score takes values in the range (0, 1); higher F1 values correspond to better model performance.

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (7)$$

The mean-square error (MSE) evaluates the mean of the quadratic prediction errors; lower values indicate better performance.

$$\text{MSE} = \frac{1}{N_{\text{pixels}}} (\text{imgref} - \text{imgout})^2, \quad (8)$$

The Matthews correlation coefficient (MCC) effectively and reliably measures the quality of binary classifications [83], even if the classes are unbalanced, by taking into account true and false positives and negatives. MCC values range from the worst value  $-1$  to the best value  $+1$ .

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}, \quad (9)$$

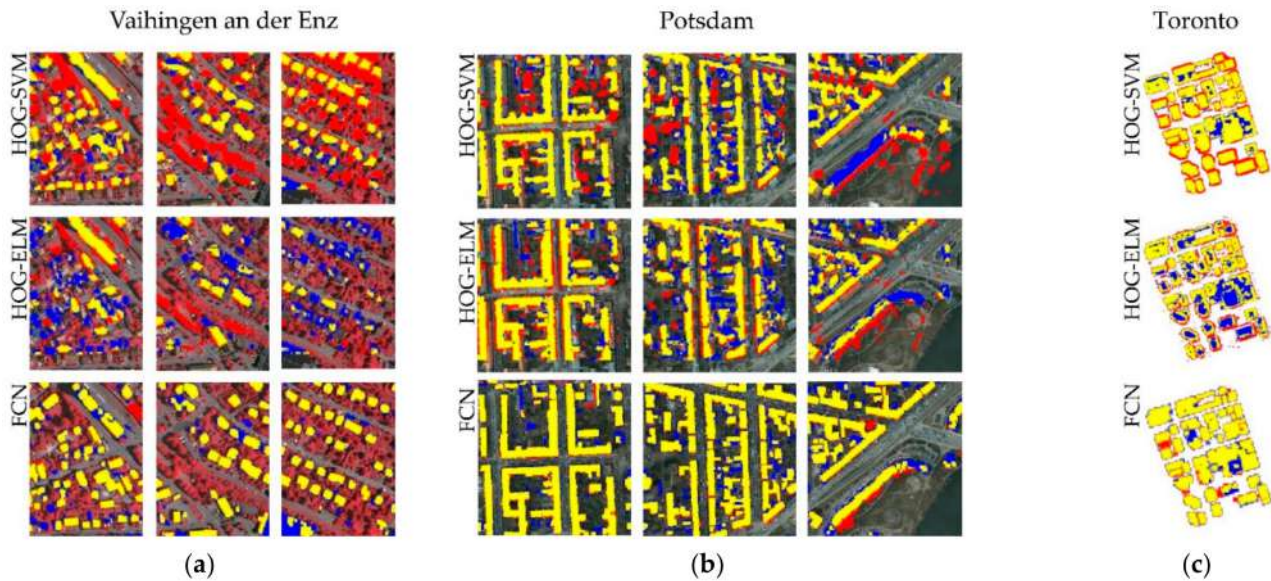
### 3. Results

We reported pixel-based metrics to evaluate the prediction error for each of the presented models for the building detection task in different urban areas.

Figure 6 shows the contingency maps resulting from the pixel-wise results of the binary classification with the SVM, EML, and FCN superimposed on the aerial images of the considered areas: the building pixels correctly detected (TPs) are colored in yellow, FPs are colored in red, and FNs are colored in blue. As seen from examples in Figure 6, ELM tended to underestimate the pixels belonging to the building class, since a fairly high rate of FNs was produced. The resulting segmentation images were not close to reality in the Vaihingen and Toronto patches: in the first case, the smaller footprints were undetected; in the second case, there were “holes” in some footprints, having building pixels surrounding non-building areas, which is a characteristic of urban fabrics formed by specific architectural typologies (e.g., courtyard, siheyuan, and patio buildings). However, the SVM seemed in the Vaihingen case more prone to overestimating the number of



building pixels due to the higher occurrence of FPs. The contingency maps produced by the FCN model showed a better correspondence of the predicted footprint position and size to ground-truth masks in all subsets.



**Figure 6.** Contingency maps obtained from a pixel-to-pixel comparison between the masks outputted by the models and the ground truth by urban area: (a) Vaihingen test set, (b) Potsdam test set, and (c) Toronto test set. True positives are shown in yellow, false positives in red, and false negatives (missed building pixels) in blue.

For quantitative evaluation of the different classifiers, we reported pixel-wise classification accuracy (Table 1) in terms of sensitivity, specificity, precision, the NPV, F1 scores, the MSE, the MCC, and the AUC using both the aggregate metrics and the metrics split by area.

**Table 1.** Pixel-wise performance by area for the three models: test evaluation metrics.

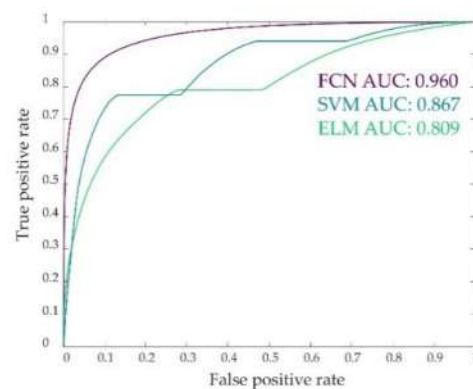
		Sensitivity	Specificity	Precision	NPV	F1 Score	MSE	MCC	AUC
HOG-SVM	Vaihingen	0.773	0.767	0.446	0.933	0.566	0.044	0.657	0.782
	Potsdam	0.737	0.876	0.718	0.886	0.728	0.079	0.620	0.828
	Toronto	0.887	0.871	0.742	0.949	0.808	0.033	0.826	0.914
	Total	0.817	0.854	0.687	0.923	0.746	0.058	0.725	0.867
HOG-ELM	Vaihingen	0.415	0.935	0.608	0.868	0.493	0.114	0.513	0.710
	Potsdam	0.720	0.888	0.735	0.881	0.727	0.084	0.620	0.840
	Toronto	0.654	0.909	0.749	0.863	0.698	0.102	0.644	0.771
	Total	0.653	0.906	0.731	0.870	0.690	0.099	0.626	0.809
FCN	Vaihingen	0.741	0.966	0.872	0.923	0.801	0.061	0.833	0.948
	Potsdam	0.801	0.948	0.784	0.899	0.792	0.060	0.834	0.964
	Toronto	0.834	0.947	0.877	0.925	0.855	0.052	0.821	0.973
	Total	0.785	0.956	0.883	0.914	0.831	0.063	0.834	0.960

Considering the Vaihingen test set, MCC values of the HOG-SVM and HOG-ELM models (0.657 and 0.513, respectively) were significantly lower than the value of the FCN model (0.833); compared with the ELM classifier, the SVM was more prone to detecting building pixels as it disclosed higher sensitivity (77.3% vs. 41.5%), but the precision was lower (44.6% vs. 60.8%). The metrics relative to non-building pixels were not found to differ markedly. Considering the Potsdam test set, the HOG-SVM and HOG-ELM models showed similarly better outcomes in terms of sensitivity but were slightly outperformed by the FCN model; the difference became greater considering the MCC, a metric that provides a balanced

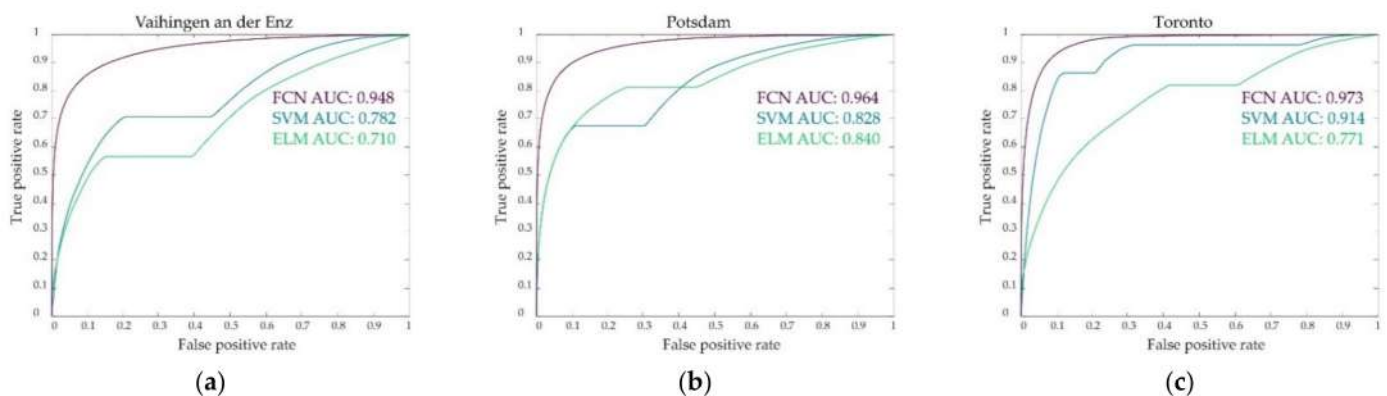
measure of the relationship between reality and prediction. Considering the Toronto test set, however, the MCC values of the HOG-SVM and FCN models were significantly higher than the value of the HOG-ELM model (0.826 and 0.821 vs. 0.664), which suffered from lower sensitivity and NPV due to the presence of small clusters of FN pixels inside the clusters of pixels representing the building footprints, as shown in Figure 6.

Evaluating the aggregated metrics, the best values were achieved by the FCN-based classifier: the FCN-based building detection method exhibited good overall detection reliability. It was found that such an approach produces good-quality results in the different urban contexts considered, as demonstrated by the values of the F1 score (0.831), AUC (0.960), and MCC (0.834), which are fairly close to the ideal value of 1.

Figures 7 and 8 illustrate the detection abilities of the three binary classifiers comparing ROC curves calculated on the output scores. The ROC curves of the SVM-based (blue lines), ELM-based (red lines), and FCN-based (orange line) classifiers plotted the dependency of the true-positive rate (sensitivity) on the false-positive rate (1—specificity) obtained at various thresholds.



**Figure 7.** Receiver operating characteristic (ROC) curves for the aggregated data, showing the relationship between true-positive and false-positive rates estimated by the three models compared to the ground truth.



**Figure 8.** Receiver operating characteristic (ROC) curves of the three models for the different urban areas: (a) Vaihingen, (b) Potsdam, and (c) Toronto.

Globally, all the final models could successfully detect the building footprints within the study areas using only DSM patches as input, presenting good predictive ability with  $AUC > 0.8$  [84].

The SVM-based classifier produced accurate predictions in Toronto ( $AUC = 0.96$ ), fairly good predictions in Potsdam ( $AUC = 0.83$ ), but minor predictions in Vaihingen ( $AUC = 0.78$ ) patches. The ELM-based classifier performance varied according to the considered urban area: the prediction accuracy on the Potsdam test set slightly surpassed the SVM accuracy ( $AUC = 0.84$  vs.  $0.83$ ), whereas the test results on Vaihingen and Toronto

datasets were the poorest among the considered methods, albeit being still acceptable (AUC = 0.71 and 0.77). The FCN curves were close to the “perfect” classifier, with AUC values between 0.95 and 0.97 for each urban area.

#### 4. Discussion

The study compared the performance of three common supervised ML techniques—HOG with SVM, HOG with ELM, and FCN—for the task of building footprint extraction from high-resolution DSM data on a benchmark dataset of DSMs of three different urban contexts (Vaihingen an der Enz, Potsdam, and Toronto). Our results showed that all the ML techniques could successfully complete the task, but the FCN appeared more robust to urban fabric diversity.

The performance of both HOG-based models was influenced by the urban context investigated and in particular decreased within the Vaihingen area. This area is more challenging due to the complexity of the urban fabrics that are composed of many small buildings varying in shape, sparse low vegetation and trees, and irregular narrow road networks. The improvements in the Potsdam area could be attributed to the larger footprints of the constructions and the relatively reduced number of trees. The worst generalization ability of the two methods based on classical feature learning and classifier training may be caused by the size of the training dataset [72,85], confirming the worse performance of the ELM classifier compared to the SVM on small datasets [72].

In summary, these results show that a DL approach based on FCNs is the most preferable method as it achieves good classification regardless of the urban context, despite the inaccuracy in contours and boundaries that is a drawback of DL-based segmentation [86], and support evidence from previous studies [35,36,38,39,41,85]. Furthermore, such an approach automates the feature engineering and benefits from transfer learning that limits the impacts of the training data size.

However, with a limited data size, although sufficient for the comparison, caution must be applied, as the findings might not be fully applicable in the case of large datasets or urban scenarios with substantially different morphological structures, e.g., Asian cities [87].

Table 2 summarizes the main advantages and disadvantages of the three methods.

**Table 2.** Advantages and disadvantages of the three implemented methods.

Method	Algorithm	Advantages	Disadvantages
HOG-SVM	[60]	- Good predictive abilities	- Training data size sensitivity - Manual feature engineering - Not suitable for urban context with sparse vegetation, small footprints, and narrow roads
HOG-ELM	[70]	- Average predictive abilities - Fastest computational time	- Possible training data size sensitivity - Manual feature engineering - Not suitable for urban context with sparse vegetation, small footprints, and narrow roads - Not suitable for urban context with large footprints
FCN	[50]	- Best predictive abilities in all urban contexts - Transfer learning (training data size robustness) - Automatic feature extraction - Easy implementation - High transferability potential	- Irregular footprint edges - Irrelevant feature susceptibility

## 5. Conclusions

In this paper, we performed an empirical comparison of three different supervised ML-based building detection methods—HOG-SVM, HOG-ELM, and FCN models—on a benchmark high-resolution remotely sensed dataset using only raster DSMs. Two of these methods belong to the classical feature learning and classifier training category—shallow ML—whereas the third is a DL network. We used HOG as a feature descriptor and trained the classifiers (SVM and ELM) for the first two methods using publicly available ISPRS datasets. The same data were employed for fine-tuning the FCN architecture through transfer learning. The high quality of the publicly available data resolved the major problem of correctly labeling the training data, on which depend the predictive skills of the final trained models. The methods are easy to implement, as analogous functions are widely accessible in both proprietary and open source software and programming languages (e.g., MATLAB, Python, R).

Our results demonstrate that determining the footprint of buildings from remotely sensed data can produce different results, depending not only on the urban morphology of the context to be surveyed but also on the model choice.

The performances of building detection techniques based on shallow ML were affected by the complexity of the urban context considered, in particular by the presence of vegetation and smaller footprints. The FCN-based model has proven to be the most robust and best-performing method for building extraction from high-resolution DSM data. Furthermore, this DL technique can generate accurate building masks without any manually engineered features with high transferability potential. Due to this, the model has the potential to solve similar pixel classification tasks, such as extraction of a different class (e.g., ground surfaces, vegetation, cars) or multi-class segmentation by being re-trained with adequate ground-truth masks and classes number. Using solely a DSM as input data increases portability between urban areas as such data are widely available, continuously improved, and constantly released [88].

Future work could investigate the potentials in the aforementioned multi-class problem domains, i.e., multi-class semantic segmentation of urban areas, as well as a systematic analysis of the impacts on the accuracy of the raster resolution variation for applications in data-poor environments or at a larger scale. To better explore the robustness to urban fabric heterogeneity and geographic transferability, future experiments may also include the application in different kinds of settlements (e.g., informal, regional, and vernacular settlements) that can present different morphological characteristics. A metrics ensemble based on both raster resolution and urban morphology could also lead to a more complete characterization of the advantages and disadvantages of each algorithm. The knowledge of the urban context, analysis of urban objects, color information from orthophotos [34], and additional data could be used to develop novel methods for classification, pre-processing the inputs, or post-processing the outputs. Thus, studies on combinations of different approaches and data to improve the FCN detection accuracy would be worthwhile.

**Author Contributions:** Conceptualization, R.A.; methodology, R.A.; software, R.A.; validation, N.M.N. and A.M.; formal analysis, N.M.N.; investigation, A.M. and N.M.N.; data curation, R.A.; writing—original draft preparation, N.M.N., A.M., R.A. and A.S.; writing—review and editing, N.M.N., A.M., R.A. and A.S.; visualization, A.M. and N.M.N.; supervision, R.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The used datasets were created by the ISPRS in the “Test Project on Urban Classification, 3-D Building Reconstruction, and Semantic Labeling” and can be required at <https://www2.isprs.org/commissions/comm2/wg4/benchmark/> (accessed on 8 October 2019).

**Conflicts of Interest:** The authors declare no conflict of interest.



## References

1. Demir, I.; Koperski, K.; Lindenbaum, D.; Pang, G.; Huang, J.; Basu, S.; Hughes, F.; Tuia, D.; Raskar, R. DeepGlobe 2018: A Challenge to Parse the Earth through Satellite Images. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 172–17209.
2. Li, L.; Liang, J.; Weng, M.; Zhu, H. A Multiple-Feature Reuse Network to Extract Buildings from Remote Sensing Imagery. *Remote Sens.* **2018**, *10*, 1350. [[CrossRef](#)]
3. Samela, C.; Albano, R.; Sole, A.; Manfreda, S. A GIS Tool for Cost-Effective Delineation of Flood-Prone Areas. *Comput. Environ. Urban Syst.* **2018**, *70*, 43–52. [[CrossRef](#)]
4. Sole, A.; Giosa, L.; Albano, R.; Cantisani, A. The laser scan data as a key element in the hydraulic flood modelling in urban areas. In Proceedings of the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences—(ISPRS Archive), London, UK, 29–31 May 2013; Volume XL-4/W1, pp. 65–70. [[CrossRef](#)]
5. Tian, J.; Cui, S.; Reinartz, P. Building Change Detection Based on Satellite Stereo Imagery and Digital Surface Models. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 406–417. [[CrossRef](#)]
6. Li, W.; Dong, R.; Fu, H.; Yu, L. Large-Scale Oil Palm Tree Detection from High-Resolution Satellite Images Using Two-Stage Convolutional Neural Networks. *Remote Sens.* **2018**, *11*, 11. [[CrossRef](#)]
7. Zhang, B.; Wang, C.; Shen, Y.; Liu, Y. Fully Connected Conditional Random Fields for High-Resolution Remote Sensing Land Use/Land Cover Classification with Convolutional Neural Networks. *Remote Sens.* **2018**, *10*, 1889. [[CrossRef](#)]
8. Audebert, N.; Le Saux, B.; Lefèvre, S. Segment-before-Detect: Vehicle Detection and Classification through Semantic Segmentation of Aerial Images. *Remote Sens.* **2017**, *9*, 368. [[CrossRef](#)]
9. Tang, T.; Zhou, S.; Deng, A.; Lei, L.; Zou, H. Arbitrary-Oriented Vehicle Detection in Aerial Imagery with Single Convolutional Neural Networks. *Remote Sens.* **2017**, *9*, 1170. [[CrossRef](#)]
10. Xu, Y.; Xie, Z.; Feng, Y.; Chen, Z. Road Extraction from High-Resolution Remote Sensing Imagery Using Deep Learning. *Remote Sens.* **2018**, *10*, 1461. [[CrossRef](#)]
11. Guo, B.; Huang, X.; Zhang, F.; Sohn, G. Classification of Airborne Laser Scanning Data Using JointBoost. *ISPRS J. Photogramm. Remote Sens.* **2015**, *100*, 71–83. [[CrossRef](#)]
12. Bretar, F. Feature Extraction from LiDAR Data in Urban Areas. In *Topographic Laser Ranging and Scanning*; Shan, J., Toth, C.K., Eds.; CRC Press: Boca Raton, FL, USA, 2017; pp. 403–420. ISBN 978-1-315-21956-1.
13. Rottensteiner, F.; Sohn, G.; Gerke, M.; Wegner, J.D.; Breitkopf, U.; Jung, J. Results of the ISPRS Benchmark on Urban Object Detection and 3D Building Reconstruction. *ISPRS J. Photogramm. Remote Sens.* **2014**, *93*, 256–271. [[CrossRef](#)]
14. Cheng, G.; Han, J. A Survey on Object Detection in Optical Remote Sensing Images. *ISPRS J. Photogramm. Remote Sens.* **2016**, *117*, 11–28. [[CrossRef](#)]
15. Kass, M.; Witkin, A.; Terzopoulos, D. Snakes: Active Contour Models. *Int. J. Comput. Vis.* **1988**, *1*, 321–331. [[CrossRef](#)]
16. Bypina, S.K.; Rajan, K.S. Semi-Automatic Extraction of Large and Moderate Buildings from Very High-Resolution Satellite Imagery Using Active Contour Model. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1885–1888.
17. Kabolizade, M.; Ebadi, H.; Ahmadi, S. An Improved Snake Model for Automatic Extraction of Buildings from Urban Aerial Images and LiDAR Data. *Comput. Environ. Urban Syst.* **2010**, *34*, 435–441. [[CrossRef](#)]
18. Liasis, G.; Stavrou, S. Building Extraction in Satellite Images Using Active Contours and Colour Features. *Int. J. Remote Sens.* **2016**, *37*, 1127–1153. [[CrossRef](#)]
19. Sun, Y.; Zhang, X.; Zhao, X.; Xin, Q. Extracting Building Boundaries from High Resolution Optical Images and LiDAR Data by Integrating the Convolutional Neural Network and the Active Contour Model. *Remote Sens.* **2018**, *10*, 1459. [[CrossRef](#)]
20. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
21. Notarangelo, N.M.; Hirano, K.; Albano, R.; Sole, A. Transfer Learning with Convolutional Neural Networks for Rainfall Detection in Single Images. *Water* **2021**, *13*, 588. [[CrossRef](#)]
22. O'Mahony, N.; Campbell, S.; Carvalho, A.; Harapanahalli, S.; Hernandez, G.V.; Krpalkova, L.; Riordan, D.; Walsh, J. Deep Learning vs. Traditional Computer Vision. In *Advances in Computer Vision*; Arai, K., Kapoor, S., Eds.; Advances in Intelligent Systems and Computing; Springer International Publishing: Cham, Switzerland, 2020; Volume 943, pp. 128–144. ISBN 978-3-030-17794-2.
23. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; IEEE: Piscataway, NJ, USA, 2005; Volume 1, pp. 886–893.
24. Konstantinidis, D.; Stathaki, T.; Argyriou, V.; Grammalidis, N. A Probabilistic Feature Fusion for Building Detection in Satellite Images. In Proceedings of the 10th International Conference on Computer Vision Theory and Applications, Berlin, Germany, 11–14 March 2015; SCITEPRESS—Science and Technology Publications: Setúbal, Portugal, 2015; pp. 205–212.
25. Ilsever, M.; Unsalan, C. Building Detection Using HOG Descriptors. In Proceedings of the 2013 6th International Conference on Recent Advances in Space Technologies (RAST), Istanbul, Turkey, 12–14 June 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 115–119.
26. Cheng, G.; Han, J.; Zhou, P.; Guo, L. Scalable Multi-Class Geospatial Object Detection in High-Spatial-Resolution Remote Sensing Images. In Proceedings of the 2014 IEEE Geoscience and Remote Sensing Symposium, Quebec City, QC, Canada, 13–18 July 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 2479–2482.

27. Tuermer, S.; Kurz, F.; Reinartz, P.; Stilla, U. Airborne Vehicle Detection in Dense Urban Areas Using HoG Features and Disparity Maps. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2013**, *6*, 2327–2337. [[CrossRef](#)]
28. Turker, M.; Koc-San, D. Building Extraction from High-Resolution Optical Spaceborne Images Using the Integration of Support Vector Machine (SVM) Classification, Hough Transformation and Perceptual Grouping. *Int. J. Appl. Earth Obs. Geoinf.* **2015**, *34*, 58–69. [[CrossRef](#)]
29. Krupinski, M.; Lewiński, S.; Malinowski, R. One Class SVM for Building Detection on Sentinel-2 Images. In Proceedings of the Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments, Wilga, Poland, 6 November 2019; Romaniuk, R.S., Linczuk, M., Eds.; SPIE: Bellingham, WA, USA, 2019; p. 6.
30. Mountrakis, G.; Im, J.; Ogole, C. Support Vector Machines in Remote Sensing: A Review. *ISPRS J. Photogramm. Remote Sens.* **2011**, *66*, 247–259. [[CrossRef](#)]
31. Lari, Z.; Ebadi, H. Automated Building Extraction from High-Resolution Satellite Imagery Using Spectral and Structural Information Based on Artificial Neural Networks. In Proceedings of the International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Hannover, Germany, 29 May–1 June 2007; Volume 36.
32. Albano, R.; Samela, C.; Crăciun, I.; Manfreda, S.; Adamowski, J.; Sole, A.; Sivertun, Å.; Ozunu, A. Large Scale Flood Risk Mapping in Data Scarce Environments: An Application for Romania. *Water* **2020**, *12*, 1834. [[CrossRef](#)]
33. Wang, P.; Zhang, X.; Hao, Y. A Method Combining CNN and ELM for Feature Extraction and Classification of SAR Image. *J. Sens.* **2019**, *2019*, 1–8. [[CrossRef](#)]
34. Nahhas, F.H.; Shafri, H.Z.M.; Sameen, M.I.; Pradhan, B.; Mansor, S. Deep Learning Approach for Building Detection Using LiDAR–Orthophoto Fusion. *J. Sens.* **2018**, *2018*, 1–12. [[CrossRef](#)]
35. Maltezos, E.; Doulamis, A.; Doulamis, N.; Ioannidis, C. Building Extraction from LiDAR Data Applying Deep Convolutional Neural Networks. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 155–159. [[CrossRef](#)]
36. Vakalopoulou, M.; Karantzalos, K.; Komodakis, N.; Paragios, N. Building Detection in Very High Resolution Multispectral Data with Deep Learning Features. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1873–1876.
37. Diakogiannis, F.I.; Waldner, F.; Caccetta, P.; Wu, C. ResUNet-a: A Deep Learning Framework for Semantic Segmentation of Remotely Sensed Data. *ISPRS J. Photogramm. Remote Sens.* **2020**, *162*, 94–114. [[CrossRef](#)]
38. Zhong, Z.; Li, J.; Cui, W.; Jiang, H. Fully Convolutional Networks for Building and Road Extraction: Preliminary Results. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1591–1594.
39. Xu, Y.; Wu, L.; Xie, Z.; Chen, Z. Building Extraction in Very High Resolution Remote Sensing Imagery Using Deep Learning and Guided Filters. *Remote Sens.* **2018**, *10*, 144. [[CrossRef](#)]
40. Fu, G.; Liu, C.; Zhou, R.; Sun, T.; Zhang, Q. Classification for High Resolution Remote Sensing Imagery Using a Fully Convolutional Network. *Remote Sens.* **2017**, *9*, 498. [[CrossRef](#)]
41. Maggiori, E.; Tarabalka, Y.; Charpiat, G.; Alliez, P. Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 645–657. [[CrossRef](#)]
42. Shrestha, S.; Vanneschi, L. Improved Fully Convolutional Network with Conditional Random Fields for Building Extraction. *Remote Sens.* **2018**, *10*, 1135. [[CrossRef](#)]
43. Audebert, N.; Saux, B.L.; Lefèvre, S. Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-Scale Deep Networks. *arXiv* **2016**, arXiv:1609.06846.
44. Arefi, H.; Reinartz, P. Building Reconstruction Using DSM and Orthorectified Images. *Remote Sens.* **2013**, *5*, 1681–1703. [[CrossRef](#)]
45. Protopapadakis, E.; Doulamis, A.; Doulamis, N.; Maltezos, E. Stacked Autoencoders Driven by Semi-Supervised Learning for Building Extraction from near Infrared Remote Sensing Imagery. *Remote Sens.* **2021**, *13*, 371. [[CrossRef](#)]
46. Chorowski, J.; Wang, J.; Zurada, J.M. Review and Performance Comparison of SVM- and ELM-Based Classifiers. *Neurocomputing* **2014**, *128*, 507–516. [[CrossRef](#)]
47. Huang, G.B.; Zhou, H.; Ding, X.; Zhang, R. Extreme Learning Machine for Regression and Multiclass Classification. *IEEE Trans. Syst. Man Cybern. B* **2012**, *42*, 513–529. [[CrossRef](#)] [[PubMed](#)]
48. Yang, L.; Lyu, K.; Li, H.; Liu, Y. Building Climate Zoning in China Using Supervised Classification-Based Machine Learning. *Build. Environ.* **2020**, *171*, 106663. [[CrossRef](#)]
49. Khan, M.A. HCRNNIDS: Hybrid Convolutional Recurrent Neural Network-Based Network Intrusion Detection System. *Processes* **2021**, *9*, 834. [[CrossRef](#)]
50. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 3431–3440.
51. Kotsiantis, S. Supervised Machine Learning: A Review of Classification Techniques. *Informatica* **2007**, *31*, 249–268.
52. Minaee, S.; Boykov, Y.; Porikli, F.; Plaza, A.; Kehtarnavaz, N.; Terzopoulos, D. Image Segmentation Using Deep Learning: A Survey. *arXiv* **2020**, arXiv:2001.05566.
53. Rottensteiner, F.; Sohn, G.; Jung, J.; Gerke, M.; Baillard, C.; Benitez, S.; Breitkopf, U. The ISPRS Benchmark on Urban Object Classification and 3d Building Reconstruction. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2012**, *1*, 293–298. [[CrossRef](#)]

54. Champion, N.; Rottensteiner, F.; Matikainen, L.; Liang, X.; Hyypä, J.; Olsen, B. A Test of Automatic Building Change Detection Approaches. In Proceedings of the CMRT09, Paris, France, 3–4 September 2009.
55. Kaartinen, H.; Hyypä, J.; Gülch, E.; Vosselman, G.; Hyypä, H.; Matikainen, L.; Hofmann, A.; Mäder, U.; Persson, Å.; Söderman, U.; et al. Accuracy of 3D City Models: EuroSDR Comparison. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2005**, *36*, 227–232.
56. Mayer, H.; Hinz, S.; Bacher, U.; Baltsavias, E. A Test of Automatic Road Extraction Approaches. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2006**, *36*, 209–214.
57. Cramer, M. The DGPF-Test on Digital Airborne Camera Evaluation Overview and Test Design. *PFG Photogramm. Fernerkund. Geoinf.* **2010**, *2010*, 73–82. [[CrossRef](#)]
58. Haala, N.; Cramer, M.; Jacobsen, K. The German Camera Evaluation Project-Results from the Geometry Group. In *Proceedings of the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences: [2010 Canadian Geomatics Conference and Symposium of Commission I, ISPRS Convergence in Geomatics-Shaping Canada's Competitive Landscape]*, Calgary, AB, Canada, 15–18 June 2010; Nr. Part 1; Copernicus GmbH: Göttingen, Germany, 2010; Volume 38.
59. Albano, R. Investigation on Roof Segmentation for 3D Building Reconstruction from Aerial LIDAR Point Clouds. *Appl. Sci.* **2019**, *9*, 4674. [[CrossRef](#)]
60. Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, 1st ed.; Cambridge University Press: Cambridge, UK, 2000; ISBN 978-0-521-78019-3.
61. Durst, N.J.; Sullivan, E.; Huang, H.; Park, H. Building Footprint-Derived Landscape Metrics for the Identification of Informal Subdivisions and Manufactured Home Communities: A Pilot Application in Hidalgo County, Texas. *Land Use Policy* **2021**, *101*, 105158. [[CrossRef](#)]
62. Foody, G.M.; Mathur, A. A Relative Evaluation of Multiclass Image Classification by Support Vector Machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1335–1343. [[CrossRef](#)]
63. Huang, C.; Davis, L.S.; Townshend, J.R.G. An Assessment of Support Vector Machines for Land Cover Classification. *Int. J. Remote Sens.* **2002**, *23*, 725–749. [[CrossRef](#)]
64. Schölkopf, B.; Smola, A.J.; Bach, F. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*; MIT Press: Cambridge, MA, USA, 2002.
65. Guenther, N.; Schonlau, M. Support Vector Machines. *Stata J.* **2016**, *16*, 917–937. [[CrossRef](#)]
66. Vapnik, V. *The Nature of Statistical Learning Theory*; Springer Science & Business Media: Berlin, Germany, 2013.
67. Wang, L. *Support Vector Machines: Theory and Applications*; Springer Science & Business Media: Berlin, Germany, 2005; Volume 177.
68. Fan, R.-E.; Chen, P.-H.; Lin, C.-J. Working Set Selection Using Second Order Information for Training Support Vector Machines. *J. Mach. Learn. Res.* **2005**, *6*, 1889–1918. [[CrossRef](#)]
69. Ertuğrul, Ö.F.; Kaya, Y. A Detailed Analysis on Extreme Learning Machine and Novel Approaches Based on ELM. *Am. J. Comput. Sci. Eng.* **2014**, *1*, 43–50.
70. Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Networks. In Proceedings of the 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541), Budapest, Hungary, 25–29 July 2004; IEEE: Piscataway, NJ, USA, 2004; Volume 2, pp. 985–990.
71. Huang, G.-B. An Insight into Extreme Learning Machines: Random Neurons, Random Features and Kernels. *Cogn. Comput.* **2014**, *6*, 376–390. [[CrossRef](#)]
72. Liu, X.; Gao, C.; Li, P. A Comparative Analysis of Support Vector Machines and Extreme Learning Machines. *Neural Netw.* **2012**, *33*, 58–66. [[CrossRef](#)]
73. Bischke, B.; Helber, P.; Folz, J.; Borth, D.; Dengel, A. Multi-Task Learning for Segmentation of Building Footprints with Deep Neural Networks. *arXiv* **2017**, arXiv:1709.05932.
74. Li, Y.; He, B.; Long, T.; Bai, X. Evaluation the Performance of Fully Convolutional Networks for Building Extraction Compared with Shallow Models. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 850–853.
75. Liu, T.; Abd-Elrahman, A. An Object-Based Image Analysis Method for Enhancing Classification of Land Covers Using Fully Convolutional Networks and Multi-View Images of Small Unmanned Aerial System. *Remote Sens.* **2018**, *10*, 457. [[CrossRef](#)]
76. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [[CrossRef](#)]
77. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
78. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; Adaptive Computation and Machine LEARNING series; MIT Press: Cambridge, MA, USA, 2012; ISBN 978-0-262-01802-9.
79. Zheng, A. *Evaluating Machine Learning Models*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2015; ISBN 978-1-4920-4875-6.
80. Scarpino, S.; Albano, R.; Cantisani, A.; Mancusi, L.; Sole, A.; Milillo, G. Multitemporal SAR Data and 2D Hydrodynamic Model Flood Scenario Dynamics Assessment. *IJGI* **2018**, *7*, 105. [[CrossRef](#)]
81. Albano, R.; Mancusi, L.; Adamowski, J.; Cantisani, A.; Sole, A. A GIS Tool for Mapping Dam-Break Flood Hazards in Italy. *IJGI* **2019**, *8*, 250. [[CrossRef](#)]
82. Powers, D.M.W. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation. *arXiv* **2020**, arXiv:2010.16061.

83. Chicco, D.; Jurman, G. The Advantages of the Matthews Correlation Coefficient (MCC) over F1 Score and Accuracy in Binary Classification Evaluation. *BMC Genom.* **2020**, *21*, 6. [[CrossRef](#)]
84. Hosmer, D.W.; Lemeshow, S. *Applied Logistic Regression*; John Wiley & Sons: New York, NY, USA; Toronto, ON, Canada, 2005; ISBN 978-0-471-72214-4.
85. Liu, T.; Abd-Elrahman, A.; Morton, J.; Wilhelm, V.L. Comparing Fully Convolutional Networks, Random Forest, Support Vector Machine, and Patch-Based Deep Convolutional Neural Networks for Object-Based Wetland Mapping Using Images from Small Unmanned Aircraft System. *GISci. Remote Sens.* **2018**, *55*, 243–264. [[CrossRef](#)]
86. Yang, Y.; Zhou, X.; Liu, Y.; Hu, Z.; Ding, F. Wood Defect Detection Based on Depth Extreme Learning Machine. *Appl. Sci.* **2020**, *10*, 7488. [[CrossRef](#)]
87. Chen, T.-L.; Chiu, H.-W.; Lin, Y.-F. How Do East and Southeast Asian Cities Differ from Western Cities? A Systematic Review of the Urban Form Characteristics. *Sustainability* **2020**, *12*, 2423. [[CrossRef](#)]
88. Uuemaa, E.; Ahi, S.; Montibeller, B.; Muru, M.; Knoch, A. Vertical Accuracy of Freely Available Global Digital Elevation Models (ASTER, AW3D30, MERIT, TanDEM-X, SRTM, and NASADEM). *Remote Sens.* **2020**, *12*, 3482. [[CrossRef](#)]