ISPRS Journal of Photogrammetry and Remote Sensing 154 (2019) 70–83

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Improving public data for building segmentation from Convolutional Neural Networks (CNNs) for fused airborne lidar and image data using active contours

David Griffiths*, Jan Boehm

Department of Civil, Environmental and Geomatic Engineering, UCL, Gower Street, London WC1E 6BT, United Kingdom

ARTICLE INFO

Keywords:
Deep learning
Convolutional neural networks
Segmentation
Image processing
Lidar
Aerial

ABSTRACT

Robust and reliable automatic building detection and segmentation from aerial images/point clouds has been a prominent field of research in remote sensing, computer vision and point cloud processing for a number of decades. One of the largest issues associated with deep learning methods is the high quantity of data required for training. To help address this we present a method to improve public GIS building footprint labels by using Morphological Geodesic Active Contours (MorphGACs). We demonstrate by improving the quality of building footprint labels for detection and semantic segmentation, more robust and reliable models can be obtained. We evaluate these methods over a large UK-based dataset of 24556 images containing 169835 building instances. This is achieved by training several Mask/Faster R-CNN and RetinaNet deep convolutional neural networks. Networks are supplied with both RGB and fused RGB-lidar data. We offer quantitative analysis on the benefits of the inclusion of depth data for building segmentation. By employing both methods we achieve a detection accuracy of 0.92 (mAP@0.5) and segmentation f1 scores of 0.94 over a 4911 test images ranging from urban to rural scenes.

1. Introduction

Reliable automatic building segmentation and mapping from aerial images and lidar and has long been sought for a range of applications. These include urban planning, disaster management, city modelling, national mapping and population management. Despite rapid technological advances in the field of machine learning, highly accurate solutions that function over large areas (i.e. entire countries) and land types (i.e. urban, rural, etc.) remain unseen. There are many reasons for this including the heterogeneous nature of both the geometry and spectral properties of buildings, unpredictable scene complexity and the loss of relevant sensor data (i.e. occlusion, poor contrast, shadows and poor image perspective) (Awrangjeb et al., 2010). Furthermore, with increasingly diverse architectural designs, deriving a general solution is arguably becoming more complex. As a result of this, the research problem of detecting buildings has been extensively studied over a number of years using a range of sensing technologies such as; satellite imagery (Saeedi and Zwick, 2008), aerial imagery (Sirmacek and Unsalan, 2008) and airborne lidar scanning (ALS) (Vosselman, 2000; Kraus and Pfeifer, 2001; Akel et al., 2004). In this paper we aim to explore this problem by investigating a range of techniques for segmentation of aerial images stacked with ALS data.

Perhaps the most exciting development in machine learning for image understanding is the recent advances in the field of deep learning. This is largely driven by significant advances in the architectures of Convolutional Neural Networks (CNNs) (Krizhevsky et al., 2012; Long et al., 2015; Badrinarayanan et al., 2017; Chen et al., 2015; He et al., 2017), which are largely accredited to the pioneering work of (LeCun et al., 1989, 1998). By pooling information from neighbourhood pixels, contextual information is carried through the network, giving the ability to not only learn specific features but also learn context about the feature. This is a particularly valuable characteristic in building segmentation, as most buildings sit in similar situations (i.e. surrounded by gardens, driveways, pavements, etc.). A key ingredient for any supervised deep learning approach is training data. Publicly available remote-sensing and Geographical Information Systems (GIS) data exists in large quantities in many countries, with the data usually being made available by national mapping agencies. The quality of open national datasets is not usually high with many errors often being present within the dataset. These errors tend to be caused by missed features, over-generalisation and in-accurately placed boundaries. Furthermore, whilst best efforts can be made to match datasets, there

* Corresponding author.
E-mail addresses: david.griffiths.16@ucl.ac.uk (D. Griffiths), j.boehm@ucl.ac.uk (J. Boehm).

https://doi.org/10.1016/j.isprsjprs.2019.05.013
Received 23 January 2019; Received in revised form 4 May 2019; Accepted 30 May 2019
Available online 06 June 2019
0924-2716/ © 2019 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS). Published by Elsevier B.V. All rights reserved.

will likely be temporal discrepancies between aerial imagery, ALS and GIS vector data. Despite this, data collection can be undertaken automatically and scales arbitrarily. The largest limitation with regard to open access GIS data is the in-ability to provide high quality segmentation ground-truth data. However, such vector data does provide a high quality object detection dataset, as object detection is only concerned with bounding box coordinates. In this paper, we demonstrate a method for automatically increasing the quality of large, low accuracy datasets to allow for training of segmentation networks. We achieve this by first calculating the center points of each building from the bounding box given by OpenMap local. This is subsequently used as the initial seed point for a morphological snake curve evolution algorithm (Álvarez et al., 2010). In addition, a differential is computed of the depth data channel which computes the edge magnitude of the ALS data. The result is strong edges on sharp geometric boundaries (i.e. roofs to ground). The snake expands such that it minimises internal and external energies along it's boundaries. Here, the most defined image structure is the edge of the building. This therefore performs a highly accurate (albeit computationally slow) segmentation of the building.

The resulting segmented images are used as training data for a Mask R-CNN. The Mask R-CNN first performs a Faster R-CNN object detection, however, continues to compute an instance segmentation within the bounding box of the Faster R-CNN prediction. This results in an overall improved semantic segmentation accuracy (and F1 score) as well as a significant increase in segmentation speed (15x). To empirically assess the value of improving training data with morphological snakes several models are trained. We achieve quantitative analysis through relative experiments where the network architecture remains consistent and only training data is altered. Moreover, we assess the potential for using RetinaNet object detection to generate initial seed points for morphological snakes to perform per pixel segmentation. All methodologies are tested on a subset of the main UK based dataset as well as on the ISPRS Potsdam benchmark dataset.

CNNs have become the de facto approach to image classification and more recently segmentation problems, however, typically these networks are concerned with only RGB colour data. Despite this, the majority of aerial imagery acquired over European and North American countries has associated depth data. This is due to the majority of aerial surveys either being acquired in a photogrammetric manner (to allow for orthorectification of the images), or acquired over landscape with available ALS data. In this paper we adopt the strategy of replacing the blue spectral band with depth data for input images. In doing this, we observe the CNN learns the radiometric and textural properties of the buildings and the inherent geometry associated with each building with respect to both the rest of the building and within the context of its surrounding.

The key contributions of our paper are: (1) presenting a new method to refine publicly available course building outline labels with morphological geodesic active contours, (2) quantifying the relative network performance increase when high quality labels are provided, (3) demonstrating the suitability for detection + refinement networks such as Mask R-CNN for building segmentation. Moreover, we provide quantitative analysis demonstrating the benefit of the inclusion of depth data for building segmentation. Although many studies adopt a image lidar fusion approach the benefits are seldom quantified.

2. Related work

Early methods for building detection and segmentation most commonly involved the extraction of higher level 2D/3D primitives from stereo images (Huertas and Nevatia, 1988; Liow and Pavlidis, 1990). Whilst such methods demonstrated good results in favourable conditions, the techniques do not work over large areas with varying scene complexities. This was due to primitives such as line segments being overly abundant in images as well as not always being indicative of the required geometric features. Different image based segmentation methods include spectral/texture analysis. However, in airborne orthorectified photographs, only the building's roof can be seen and many roofs possess very similar spectral/texture characteristics to other common features (i.e. car parks, sidewalks, game areas, etc.). Other issues of a purely radiometric approach are caused by varying light intensities from sun strength, surface reflectivity and shadows. These issues can leave buildings to look different depending on environmental variables at the time of data acquisition. Therefore, in recent years (e.g. 1990s - present) the use of airborne laser scanning has been most commonly used as this allows for direct inference of the scenes geometry. Furthermore, research in this area has been accelerated by a large number of countries benefiting from freely available, high resolution (e.g. 25 cm–1 m) ALS data from national mapping and environmental agencies. For example, a typical approach for segmentation from 3D point clouds involves identifying planar segments as buildings are generally made up of planar surfaces (Dorninger and Pfeifer, 2008). Whilst these methods generally offer improvements to purely radiometric based approaches, they still encounter many similar and new issues. Typical issues include occluded surfaces masking or splitting up planar surfaces, outlying points within the point cloud making planar surfaces appear non-planar and 3D points not belonging to planar surfaces being excluded from the segmentation. In recent years these methods have been improved by the incorporation of novel segmentation algorithms (Hernandez and Marcotegui, 2009; Chen et al., 2014; Xiao et al., 2013) and machine learning classification methods such as Support Vector Machines (SVMs) (Secord and Zakhor, 2007) and Random Forest (RF) classifiers (Guo et al., 2011). Such limitations has therefore lead to a number of researchers, such as Strom et al. (2010), Aijazi et al. (2013), Vetrivel et al. (2015), Marmanis et al. (2018) to combine the benefits of both radiometric and geometric data for segmentation.

CNN's are currently one of the most actively researched method for building detection. Vakalopoulou et al. (2015) demonstrated that use of classifying high-resolution satellite imagery (QuickBird and WorldView 2) as either a 'building' or 'not-building'. Furthermore, the use of multispectral imagery, most noticeably near infrared (NIR), could be substituted to improve detection results. This approach used CNNs to initially classify crops of the image to act as training data for a SVM classifier. Saito and Aoki (2015) presented an end-to-end CNN approach for semantic segmentation of both roads and buildings using publicly available GIS data. A patch-based semantic segmentation approach was used, where the aerial image was first divided into smaller patches for training and classification. This achieved state-of-the-art results and demonstrated the effectiveness of including Maxout and Dropout to aerial image segmentation tasks. Whilst patch-based methods demonstrated good results (Guo et al., 2016), they suffered from limitations caused by memory inefficiencies (Ciresan et al., 2012; Kampffmeyer et al., 2016), ultimately making the approach unfeasible for large (i.e. national) datasets. A further example of the leveraging of free aerial data and large GIS datasets (OSM) is presented by Yuan (2016). A simple CNN was developed with structure integrating multi-layer information for pixel-wise classification. This enabled the network to learn hierarchical features for segmenting individual features and showed promising results for generalisation over a much larger area. Zhang et al. (2016) demonstrated a simple implementation of a CNN could be used to detect with reasonable precision (89%) Google Earth aerial imagery. A more sophisticated approach is presented by Paisitkriangkrai et al. (2015), Quang et al. (2015) who combined the input of hand-crafted features for training and conditional random fields as a post-processing step to semantically labelled images. The conditional random fields were shown to increase label confidence and incorrectly labelled pixels. Wu et al. (2018) proposed a novel multi-constraint fully Convolutional network (MC-FCN), improving the existing U-Net architecture for building segmentation in aerial images. Extra constraints were added onto the intermediate layers enabling a greater ability for feature representation and ultimately lead to a higher

performing model.

Research in the fusion of spectral and depth data into CNN training has been demonstrated in a number of studies. Socher et al. (2012) perform semantic segmentation of RGB-D images by beginning their CNN pipeline with a single convolution and pooling layer. They then stack Recursive Neural Networks (RNN) which are employed on local blocks of the previous layer in a convolutional manner. Marmanis et al. (2016) proposed a method of using an ensemble of FCNs which take an input of intensity and depth data for semantic segmentation of the ISPRS Vaihingen benchmark dataset, achieving at the time, state-of-the-art results. Marmanis et al. (2018) further developed ensembles of CNNs with a more advanced network architecture. The Digital Surface Model (DSM) was used as depth data to allow the model to learn geometrically meaningful boundaries. More specifically, the model combines semantic segmentation with semantically informed edge detection, making the class boundaries explicit in the model. Empirically including class boundaries significantly improved the CNN performance and achieved state-of-the-art results in the ISPRS Potsdam and Vaihingen benchmark datasets for multiple classes. Delassus and Giot (2018) demonstrated by fusing a DSM to the SpaceNet winning solution results could be improved between 1% and 7% for various scenes. Sun and Wang (2018) use DSMs to refine segmentation results derived from a Fully Convolutional Network (FCN) trained using very high resolution satellite images. Bittner et al. (2018) directly fuse normalised lidar, spectral and panchromatic data together for training in a 3 branch FCN network in their network FUSED-FCN4S. Chen et al. (2018) supply a true orthophoto and DSM into their Deeply-supervised Shuffling Convolutional Neural Network (DSCNN), a multi-scale extension of the Shuffling Convolutional Neural Network (SCNN). The main contribution is the proposal of fusing multi-scale features for a more rich feature descriptor. Huang et al. (2019) also use a FCN architecture but further introduce gated feature labels to allow for multi-scale feature communication in their network GRRNet. Simarly to our study the authors have chosen to drop the blue channel and instead opt for the input channels of Red, Green, near-infrared and normalised DSM. Unlike the networks noted here, we do not use a FCN architecture but instead opt for a object detection + refinement approach.

3. Methodology

3.1. Data preparation

Data collection was performed by an automated script which allowed for large data volumes to be collected and processed in a reasonably short amount of time (≈4 h). All data acquired was aerial information over England, UK. 25 cm RGB imagery is first obtained from the OS. Next, pre-computed DSMs from 50 cm airborne lidar are obtained from the UK Environment Agency and re-sampled to 25 cm. Finally, OS OpenMap local data is downloaded over the area. Building footprint shape-files are extracted and converted to raster format at 25 cm resolution. This information is downloaded by 10 km^2 OS Grid-Reference tiles. Any areas missing lidar data are removed. The data is then merged into RGB-D images and label data respectively and cropped to 250 m^2 tiles. As the OS tiles cover varying land topographies it was also necessary to normalise the depth channel between 0 and 255 for each image tile. This ensured the model would learn the contextual relationship of the building geometry in respect to its local area, and the absolute height (above a datum) is not considered. As a result a more consistent channel input is realised, with building roofs typically having pixel values between 150 and 255 depending on surrounding buildings and presence of high vegetation. Whilst saving each label tile the bounding box coordinates for each building instance were extracted and saved to.xml file in the Pascal dataset format. This was computed by running a binary image border extraction algorithm (Suzuki and Be, 1985) to obtain border locations and computing the bounding area. 9 tiles were collected covering a wide diversity of land cover, ranging from, Urban (Greater London) to rural. Tiles covering rural areas were susceptible to containing large areas where no buildings were present. This can cause an extreme foreground-background class imbalance during training as large areas in rural tiles will contain no buildings, effectively saturating the CNN with easy to detect negative examples. This has been shown to cause issues in training and be responsible for decreasing accuracy in object detectors (Lin et al., 2017). To account for this, any tiles with no buildings (wrt OS OpenMapLocal labels) is consequently deleted and not included in the model training/evaluation. After trimming, the data consisted of 24,556 images containing 169,835 instances of buildings (Table 1).

Table 1
UK data collected for model training and testing using a automated script. Tile names correspond to OS area codes for which the images are geographically located.

OS Tile	Number of Images	Building Instances
TL50	1695	6110
SZ10	4060	38,898
TF03	678	2604
TQ24	3450	19,296
TQ39	7420	53,422
TQ58	4987	42,512
TQ67	2266	6993
Total	**24,556**	**169,835**

To compare our final model against other methods discussed in Section 2 the ISPRS Potsdam[1] semantic labelling benchmark datasets were also downloaded. The Potsdam dataset contains similar RGB and lidar data to the data collected via the methods above, however, are a higher resolution (5 cm and 9 cm respectively). The Potsdam dataset is not included within the main model training dataset. Evaluation therefore measures the models ability to generalise. To quantify this we also fine-tune the OS data derived trained model with a sub-sample of the Potsdam data separately and re-evaluate the models performance.

Once the data is stored in RGB-D images we further refine our image to RGB, RG-D_E and $_N$ channels. Where the depth channel 'D' is the edge magnitude and normalised DSM layers respectively (see Fig. 1).

3.2. Morphological Geodesic Active contours

The frequent use of Active Contour Models (ACM) has been consistent in computer vision applications (i.e. shape recognition, object tracking and segmentation) since its inception (Kass et al., 1988). Geodesic Active Contours (GACs) (Caselles et al., 1997) advanced on prior work by proposing a robust technique in which ACM's evolve in time according to intrinsic geometric measures of an image (Fig. 2A). In general, a ACM can be defined as an energy-minimising two-dimensional spline curve of points $p_i = (x_i, y_i)$ for $i = 1, 2, \ldots, n$ where x_i and y_i are the x and y coordinates respectively and n is the number of points. To describe a classical energy base snake (Kass et al., 1988), let $C(q)$: $[0, 1] \rightarrow \mathbf{R}^2$ be parametrised planar curve and let I: $[0, a]x[0, b] \rightarrow \mathbf{R}^+$ be a given area with edges which we want to segment, the energy of curve C is given by:

$$E(C) = \alpha \int_0^1 |C'(q)^2 dq + \beta \int_0^1 |C''(q)|^2 dq - \lambda \int_0^1 |\nabla I(C(q))| dq \quad (1)$$

where α, β and λ are real positive constants. Here the first two terms are responsible for controlling the smoothness of the contours, and the third for attracting the contour towards the edges in the image (external energy). Therefore, solving the ACM problem amounts to finding, for a given set of constants α, β, and λ and the curve C that minimises E.

GACs differ to classic parametric ACMs in their ability to naturally

[1] http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-potsdam.html.

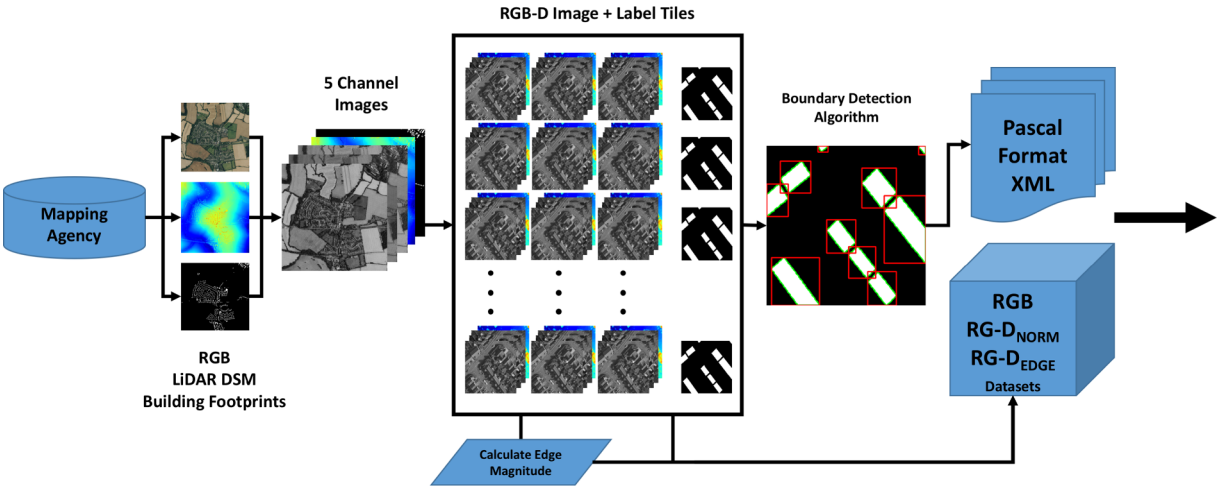

Fig. 1. Data preparation workflow for improved label segmentation and model training.

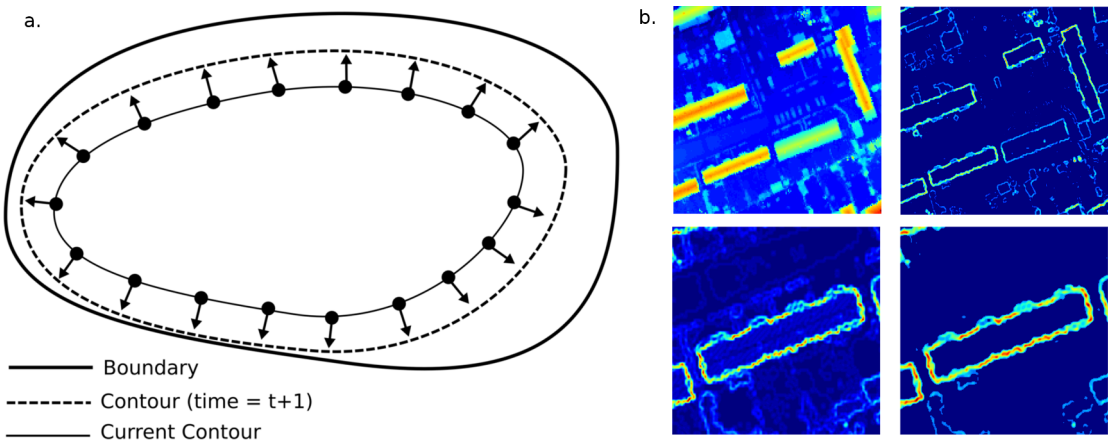

Fig. 2. (A) Typical example of an ACM expanding towards a boundary. (B) (i) DSM raster channel in image after normalisation between 0 and 255, (ii) Edge magnitude of DSM after data cleaning and pre-processing, (iii) Individual building prior to pre-processing, (iv) Individual building post pre-processing.

handle changes in the topology of the curve, as well as not relying on the parameterisation of the contour. In GACs the engery function is a geodesic in a Riemannian manifold with a metric induced by image features (i.e. target borders). The GAC further incorporates methods learned from euclidean curve shortening and level sets. The energy minimisation is achieved by solving Partial Differential Equations (PDEs) on an embedding function that has the contour as its zero-level set. The PDE for GACs is defined as:

$$\frac{\delta u}{\delta t} = g(I)|\nabla u|\mathrm{div}\left(\frac{\nabla u}{|\nabla u|}\right) + g(I)|\nabla u|\nu + \nabla g(I)\nabla u \tag{2}$$

where g: $[0, +\infty[\leftarrow \mathbf{R}^{+}]]$ is a strictly decreasing function such that $g(r) \leftarrow 0$ as $r \leftarrow \infty$ and u and ν are signed distance functions. The first term is the smoothing term, the second is the balloon term and the third is the image attachment term.

The GAC also employs the concept of a balloon force as first proposed by Cohen (1991). The balloon factor makes the curve behave more like a balloon being inflated by an additional force. This ensures the contour is not stopped on weak edges as well as allowing the contour to hold a degree of integrity to its current shape. This is particularly beneficial in our case as the lidar data used is not very high quality (50 cm). The coarseness of the data causes apparent gaps in strong edges caused by partial hits of sharp geometric edges of buildings. The result is a strong edge with intermittent low intensity pixels. The balloon factor therefore makes the contour mostly invariant to these small gaps.

The most recent advancement in ACMs is the proposal of a morphological approach (Márquez-Neila et al., 2014) which can be used to enhance GACs as well as Active contours with edges (Chan and Vese, 2001). Morphological ACMs work similarly to classical ACMs, however, instead of solving PDEs and level-sets over a floating point array, morphological operators (i.e. dilation and erosion) are used over a binary array. The morphological ACM is therefore approximating the PDEs solutions making the model faster and numerically more stable. In this paper we use an implementation of a morphological approach on a GAC (MorphGAC). The approach utilises the ability to express some morphological operator as PDEs. For example, a dilation D_h and erosion E_h with radius h of function u can be defined as:

$$D_h u(x) = \sup_{y \in hB(0,1)} u(x+y) \tag{3}$$

$$E_h u(x) = \inf_{y \in hB(0,1)} u(x+y) \tag{4}$$

where $B(0, 1)$ is a ball of radius 1 centered at 0 and hB is the set B scaled by h so that $hB = \{hx: x \in B\}$.

The dilation D_h can be used to show:

$$\lim_{h \to 0^{+}} \frac{D_h u - u}{h} = |\nabla u| \tag{5}$$

Therefore, the successive application of D_h with very small radius, $\lim_{m \to \infty}(D_{t/m})^m u_0$, is equivalent to:

$$\frac{\delta u}{\delta t} = |\nabla u| \tag{6}$$

with initial value $u(0, x) = u_0(x)$. Thus, we can say the dilation has *infinitesimal behaviour* equivalent to the PDE.

This can also be demonstrated for the erosion (E_h) function where:

$$\lim_{h \to 0^+} \frac{E_h u - u}{h} = |\nabla u| = \frac{\delta u}{\delta t} \tag{7}$$

A full explanation on morphological operators is beyond the scope of this paper and the reader is referred to the original paper (Márquez-Neila et al., 2014).

Whereas morphological active contours without edges works well without defined boundaries, MorphGACs require a strong edge. This method is choosen here as the sharp geometric shape of buildings on a DSM produce a pronounced edge of gradient change. To further exemplify the edge we take an approximation of the differential of the DSM channel on the tile to be processed. This is achieved by using the common sobel operator (Kanopoulos et al., 1988) in both horizontal (x) and vertical (y) directions. This, in essence, gives us the gradient magnitude, therefore, sharp changes in elevation have strong responses and vice versa. This offers the favourable conditions for our MorphGAC to grow given an optimum starting (seed) point. To compute the seed point, the moments are computed for each OpenMapLocal building polygon and the centre point which lies within the polygon is extracted. To clean the data, first a simple binary mask is applied with a threshold value of 20. This removes any noise on top of the building roof. Finally, a Gaussian blur is applied to the tile and a single closing morphological operator is run over the image to limit the number of gaps within the building boundaries (Fig. 3).

However, one of the largest issues with large-scale manually labelled building footprints is overgeneralisation over smaller buildings. To correct for this, before the seed point is determined for the building footprint, a multiple building check is performed. This is achieved by first masking the DSM with the footprint label. Next a k-means where $k = 5$ is computed on the image histogram. A threshold value is then located where $k = 3$ and any point with a pixel value x where $x > (k = 3)$ is defined as above ground. Individual buildings are then detected with the Douglas-Peucker algorithm. The seed points are then computed for each individual building using the contour moments (Fig. 4).

3.3. Convolutional neural network models

CNNs for image feature understanding typically fall into 3 categories; classification, object detection and semantic segmentation. Whereas object detection is concerned with localising an object within an image, semantic segmentation generates pixel-wise classifications of a given scene. There is a wide range of methods for effective object detection, however, they can generally be further categorised into one-stage and two-stage detectors. The two-stage approach was popularised by (Girshick et al., 2014) and, put simply, works by firstly generating a regional proposal for potential bounding box locations, and secondly, classifies each region proposal candidate using a classification CNN (i.e. ResNet (He et al., 2016) or VGG-16 (Simonyan and Zisserman, 2014)). The one-stage approach was popularised by Sermanet et al. (2013), Liu et al. (2016), Redmon et al. (2016) and motivated by the potential to speed up the two-stage process which has many speed limitations such as their inability to optimise or parallelise. Instead, the one-stage detector applies a dense sample of classifications over the image at various scales and aspect ratios. The classifications with the highest probabilities for containing a given object are used as the object's location within the image. This computationally cheaper method has allowed for one-stage object detectors to be deployed on fairly basic hardware (i.e. mobile phones) for real-time detection, however, usually achieve poorer accuracy than their two-stage counter-parts. Despite this, it is evident that the gap between one-stage and two-stage detectors is becoming slighter. We first assess two model architectures, RetinaNet (Lin et al., 2017) and Faster R-CNN, which are one-stage and two-stage object detection architectures respectively.

For all processing scenarios we use pre-trained weights for all three channel datasets. The weights were pre-trained through a two-step process. Firstly, the RetinaNet is trained on the ImageNet dataset (~350,000 images). This provided a foundation of very strong high level weights, however, almost all of these images are from terrestrial sources and are therefore not very relevant for aerial images. Therefore, we secondly trained the network on the dataset for Object detection in Aerial images (DOTA) dataset (Xia et al., 2018), which consisted of 2806 aerial images with 188,282 manually labelled instances. This provided more reasonable initial weights for training.

Initially, adaptations were made to both RetinaNet and Mask/Faster RCNN to allow for 4 channel inputs (RGB-Depth), however in both cases this substantially reduced the networks performance. We

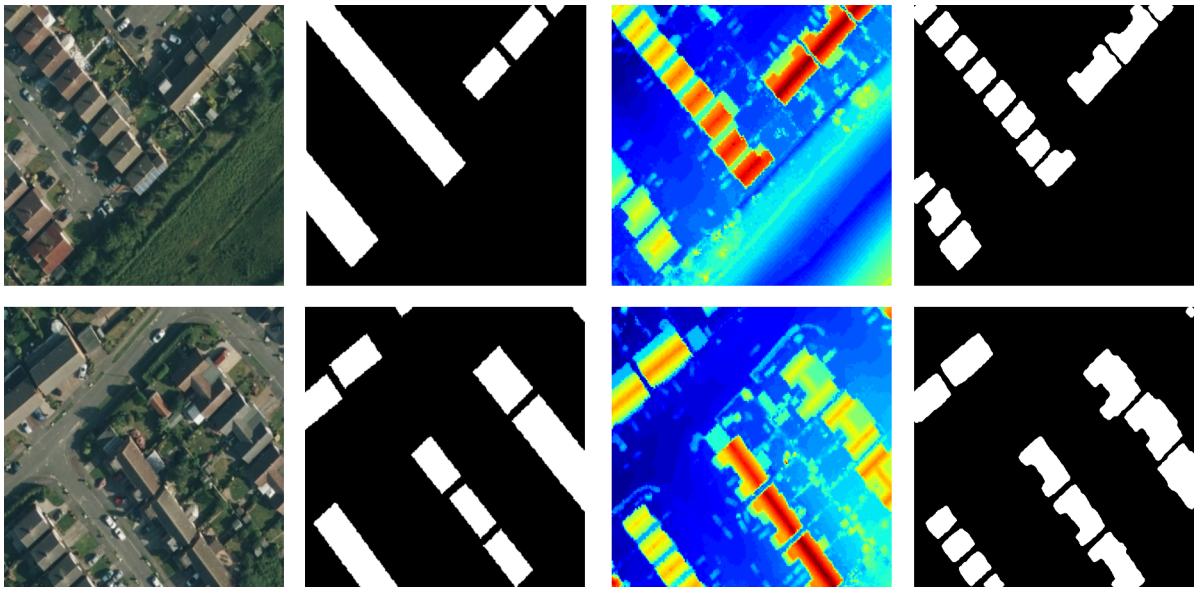

Fig. 3. Delineation of multiple buildings from over-generalised labels. An otsu binarisation threshold is utilised over each label to determine if the label has been over-generalised. If this is the case multiple geomorphological snake seeds points are extracted for the center of each individual building.

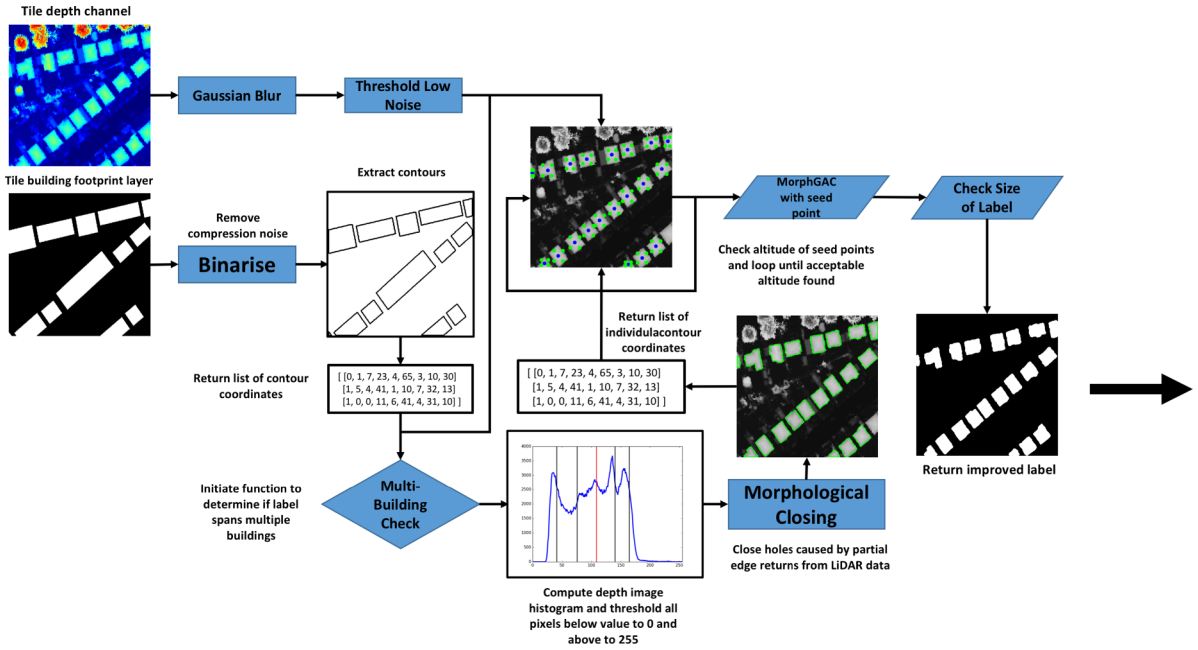

Fig. 4. Workflow for improving building footprint labels.

therefore opted to solely train on 3 channel models. This also enables existing architectures to be used much more easily without the need for network alteration. Moreover, the networks remain more lightweight. The blue channel was removed as it strongly correlated to both the red and green channels and therefore contains a large amount of redundant information. This is further justified as over both urban and rural environments in the UK blue is likely to be the least dominant and informative channel. Although, the third (blue) channel weights had no relevance to depth, we found that initialising the model with RGB pre-trained weights improved accuracy for RG-D datasets. This is likely due to low-level (top) features being generally concerned with local maximas/minimas, blobs and edges which is relevant for the detection of sharp geometric boundaries in the depth channel. We therefore find that the ImageNet and DOTA weights generalise to the depth channel for the higher layers of the network, which are often accredited to requiring the largest amount of training data. Internal comparisons were made against models initialised with random weights for all channels. Random weights were sampled around a truncated normal distribution centred on 0 with stdev $= \sqrt{\frac{2}{\eta}}$ where η is the number of input units in the weights tensor (4 in this instance) (He et al., 2015).

3.3.1. RetinaNet

The most notable advancement in recent years for one-stage object detectors is proposal of the focal loss algorithm. The focal loss aims to address the issue of large class imbalance between foreground and easy to detect background (i.e. 1:1000). This large imbalance is therefore particularly detrimental to accuracy when single class labels, like in our work, are used. This issue is addressed by re-shaping the cross-entropy loss such that it down weights the loss assigned to well-classified features during training. In our case the well-classified features refer to all image space that is not part of a building (easy negatives). The reshaped focal loss first differentiates between 'easy' and 'difficult' examples and then down weights any 'easy' examples in the training. This therefore focuses the training on the difficult examples (buildings). The focal loss starts from a Cross Entropy (CE) binary classification[2] such that:

$$CE(p, y) = \begin{cases} -\log(p) & \text{if } y = 1 \\ -\log(1-p) & \text{otherwise,} \end{cases} \tag{8}$$

Here $y \in \{\pm 1\}$ specifies the ground-truth class and $p \in [0, 1]$ is the models estimated probability for the class with label $y = 1$. This can be rewritten as:

$$p_t = \begin{cases} p & \text{if } y = 1 \\ (1-p) & \text{otherwise,} \end{cases} \tag{9}$$

where $CE(p, y) = CE(p_t) = -\log(p_t)$. The weighting is then applied by adding a modulating factor $(1 - p_t)^{\gamma}$ with tunable focusing parameter $\gamma >= 0$. The focal loss function is then defined as:

$$(p_t) = -(1-p_t)^{\gamma}\log(p_t) \tag{10}$$

For classification RetinaNet uses a feature pyramid network (FPN)[3] (Lin et al., 2017) backbone on top of a feedforward ResNet architecture. This method is used as it generates a rich, multi-scale convolutional feature pyramid. The network then attaches two sub-networks, one for classifying anchor boxes, and one for regressing from anchor boxes to ground-truth object boxes.

We train the RetinaNet model on five datasets; RGB, RG-D_E, RG-D_N, RGB-D_E and RGB-D_N. Here NORMALISED refers to depth data tiles that has been normalised between pixel values of 0–255. We also train the model with edge-magnitude data for the depth channel (EDGE). This is to assess how the model learns depth data, and more specifically, if the geometric texture of the roof aids the network training, or if the training is dominated by the strong geometric boundaries of the building instances. The network is implemented in Google's TensorFlow (Abadi et al., 2015).

In line with the original paper the drop out regularisation technique was not used in the model training. However, we do include batch normalisation within the ResNet backbone implementation. This has been demonstrated to accelerate training speed by reducing the internal covariate shift with respect to the networks training weights (Ioffe and Szegedy, 2015). Although the absence of drop out has potential to lead to over-fitting of the model, inconsistencies between the aerial image, airborne lidar and building footprint data has to ability to act as a

[2] The binary classification example is taken from the original paper and is the most relevant example in our case. Extending the focal loss to multi-class is straightforward and functional.

[3] This offers a good comparison as F-RCNN also uses a FPN.

pseudo drop out.

In each processing scenario the network was trained for 100 epochs, where an epoch is defined as a single presentation of all training data through the network. Training data was passed through the network in batches of size 8. An initial learning rate of 1e-5 was used with a decay rate of 0.96. The learning rate was therefore computed as: learning rate = learning rate $*$ decay rate$(\frac{\text{global step}}{\text{decay step}})$. The network has a total of 36,276,717 and 36,279,853 trainable parameters for the three channel and four channel architectures respectively. This was trained over the course of a week on two Nvidia 1080 Ti graphical processing units (GPUs).

To allow for an automated per-pixel segmentation using the RetinaNet, we use the centroid of the bounding box output a seed point for the MorphGAC algorithm. As in Fig. 4 the altitude is checked for this seed point and the seed iteratively moved until a suitable start point is found.

3.3.2. Faster/Mask R-CNN

Faster R-CNN (Ren et al., 2015) and Mask R-CNN (He et al., 2017) are generally considered state-of-the-art architectures for object detection and instance segmentation respectively. Both these architectures are part of the Region based CNN family with each being developed upon the prior. Faster R-CNN is effectively two networks; a region proposal network (RPN) and a Fast R-CNN detector network. The RPN replaces the more time-consuming selective search method which proved to be a bottleneck in previous versions (R-CNN and Fast R-CNN). In Faster R-CNN the RPN solves this by instead using the feature map derived from the last convolutional layer in the backbone CNN (i.e. VGG16/ResNet) as the proposals for detector network. This allows for a single CNN to be used for both the region proposals and classification. The network then performs a ROI pooling, fully connected layer and classification as in R-CNN. The RPN makes use of anchors to generate sensible potential bounding boxes along with scores that determine how likely each box is to contain an object. Initial proposal locations are determined by the inclusion of *anchors*. An anchor is a given proposal box with center x, y, defined by the current position of a sliding window over a feature map at each given stride. Let k denote the maximum number of proposals at a given location, we compute proposals at 3 aspect ratios and 3 scales, therefore, in our case (and as in the original paper) $k = 3$. Anchors with a high *objectness*[4] are passed to the RPN which outputs proposals to be processed by the classifier and regressor to predict class and class-specific box refinement. The SmoothL_1 location loss function is used for training and is defined as:

$$L(p_1, t_i) = \frac{1}{N_{\text{cls}}} \sum L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum p_i^* L_{reg}(t_i, t_i^*) \tag{11}$$

where i is the anchor index in a mini batch, p_i is the predicted probability of proposal being an object, ti is the coordinates of the predicted bounding box, L_{cls} os the log loss, p_i^* is the ground truth objectness label, L_{reg} is the SmoothL_1 loss and t_i^* is the true box coordinates. The loss function of the refressor can then be defined as:

$$\text{Smooth}L_1(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise,} \end{cases} \tag{12}$$

Mask R-CNN builds directly on Faster R-CNN by adding a third output branch for each candidate object. Whereas the prior branches were responsible for class label and bounding-box offset, the authors now output object mask proposals. This can therefore be trained in parallel with the Faster R-CNN network, outputting a binary mask for each Region of Interest (RoI). The loss function is updated to then include the mask predictions so that; $L = L_{cls} + L_{box} + L_{mask}$, where L is the total loss.

[4] Objectness is defined as the measure of membership to a set of object classes vs. background.

We implement Mask-RCNN using a ResNet backbone for classification. As with RetinaNet the training datasets include; RGB, RG $-$ $D_{NORMALISED}$ and RG $-$ D_{EDGE}. The same procedure for model pre-training is also used (as described in Section 3.3.1). The model is trained for 100 epochs, with a batch size of 8, on a single Nvidia 1080ti GPU. An initial learning rate of 3e-4. The learning rate was decreased to 3e-5 and 3e-6 at epoch 50 and 75 respectively. The network has a total of 104,112,117 trainable parameters.

3.4. Model performance

For segmentation scenarios the model performance is evaluated by computing the precision, recall and F_1 accuracy for each processing scenario. We define each metric as; $Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$, $Precision = \frac{TP}{TP + FP}$, $Recall = \frac{TP}{TP + FN}$ and $F_1 = 2 * \frac{Recall * Precision}{Recall + Precision}$, where TP, TN, FP, FN are true positive, true negative, false positive and false negative respectively for each pixel.

To evaluate the object detection accuracy of the models during training, we calculate the mean Average Precision (mAP), where a positive is defined with having an Intersection over Union (IoU) with the ground truth box >0.5. The mAP is computed as the average of the maximum precision at 11 recall vales ($r \in 0.0, 0.1, \ldots, 1.0$). The mAP can therefore be defined as:

$$AP = \frac{1}{11} \sum_{r \in 0.0, 0.1, \ldots, 1.0} AP_r \tag{13}$$

where

$$AP_r = \max_{\tilde{r} >= r}(\text{precision}(\tilde{r})) \tag{14}$$

4. Results

4.1. Mask R-CNN

Mask R-CNN (and therefore Faster R-CNN) networks are first trained for datasets containing public labels. The process was repeated for RGB, RG-D_E and RG-D_N channel images. The results are recorded in Table 2 and training statistics visualised in Fig. 5. The most prominent disparity exists between models that have been trained with a depth data channel, against the models without. All scenarios perform well (>90%) under the segmentation accuracy metric, however, the F1 is likely a much more representative metric for model performance. One reason for this is that the test dataset contained an average of 15% building pixels. The incorporation of depth data resulted in a 25% and 31% increase in F1 value for RG-D_E and RG-D_N respectively. Precision values demonstrated little variability relative to recall values. This suggests that the incorporation of depth data was most prominent in resolving false negatives. Further suggesting under-segmentation was a key issue with inference when the depth data was not present. Improvements were also seen in Faster R-CNN mAP scores, this demonstrates the Faster R-CNN object detection branch also benefited from depth data inclusion.

An overall improvement is noticed with the replacement of public labels for improved labels (discussed in Section 3.2), as seen in Table 2. In contrast to the inclusion of depth data, improved labels had a greater impact on the precision value. This therefore suggests the greatest progress was in the resolution of false positives. As one of the key issues with public building footprints is over-generalisation and thus, over-segmentation, this was anticipated. The improvement of recall resulted in a 21%, 14% and 9% increase against their public label counterparts for RGB, RG-D_E and RG-D_N respectively.

The difference between RG-D_E RG-D_N is minimal, with converging mAP and F1 scores being almost identical. The most noticeable difference is delay in convergence of RG-D_E for both loss and mAP with respect to RG-D_N (Fig. 5). This suggests that whilst the normalised depth

Table 2
Mask R-CNN segmentation results. mAP is calculated for recall >0.5 with true positives defined as IoU >0.5.

Model	mAP	Precision	Recall	Accuracy (%)	F1 Value
RGB OS labels	0.45	0.73	0.61	91.33	0.61
RG-D_E OS labels	0.54	0.76	0.79	94.24	0.76
RG-D_N OS labels	0.51	0.76	0.87	95.27	0.80
RGB labels +	0.68	0.82	0.72	93.97	0.74
RG-D_E labels +	0.80	**0.89**	0.88	97.09	**0.87**
RG-D_N labels +	**0.81**	0.88	**0.90**	**97.11**	**0.87**

Models trained with the improved labels demonstrated a noticeable improvement for training sets contained depth data. This most represented by the increase in recall values by 130% and 25% for RG-D_N and RG-D_E respectively. The absence of improvement in the RGB image sets suggests that the improved labels had the largest impact on the depth channel in the network training. Despite Potsdam being noticeably different to the training dataset in terms of building architecture and radiometric properties, the general model with the inclusion of depth performs as well on the Potsdam dataset as the RGB model trained with improved labels on it's standard test set. The significance

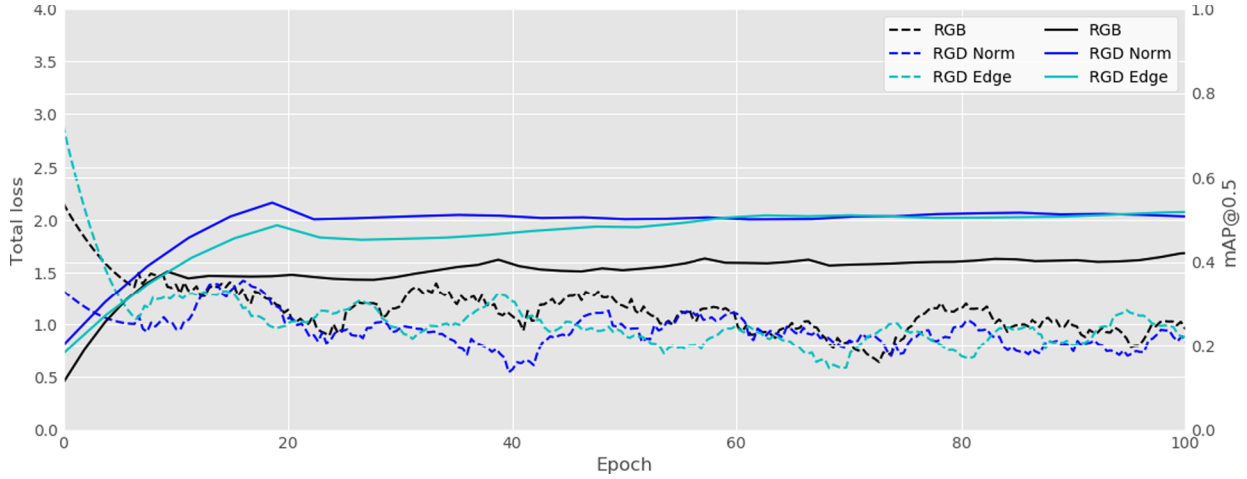

Fig. 5. Public data labels training plots for Faster R-CNN training. Datasets containing depth data demonstrate an improvement in mAP score. RG-D_E appears to learn at a slower rate, however, converges at a very similar value to RG-D_N.

channel is more similar to the blue channels of the images used for pre-training, the network is still capable of learning a unique geometric-based image channel. Furthermore, it suggests that in this instance the network does not favour a particular depth channel design. This is likely due to the most significant information being located at the building edge where a sharp change in altitude is represented by a strong edge.

The models discussed above are then applied to the Potsdam benchmark dataset (Table 3). The results observed on the Potsdam dataset are not generally consistent with the prior results on the main training dataset (Table 2). In public label scenarios the most prominent statistic is the reduction of recall value. This is a result from an increase in false negatives. As the reduction in recall is generally caused by under-segmentation of a feature, this explains the steep increase precision values. Buildings in the Potsdam dataset on average covered 22% of the image. This explains that whilst the accuracy values are respectfully high, the performance based on the F1 metric is very poor. The RG-D_N did improve the performance somewhat, however, poor recall values suggest the results are unreliable and unrepresentative of the overall performance.

Table 3
Potsdam Mask R-CNN segmentation results.

Model	mAP	Precision	Recall	Accuracy (%)	F1 Value
RGB OSLabel Generic	0.30	0.80	0.44	81.16	0.53
RG-D_E OSLabel Generic	0.34	0.88	0.30	77.42	0.41
RG-D_N OSLabel Generic	0.36	0.84	0.55	89.98	0.66
RGB Label + Generic	0.41	0.96	0.32	83.57	0.55
RG-D_E Label + Generic	0.48	0.79	0.69	86.98	0.73
RG-D_N Label + Generic	0.48	0.87	0.69	88.01	0.75
RGB Fine-Tune	0.71	0.91	0.91	95.30	0.91
RG-D_E Fine-Tune	0.78	0.92	0.92	96.03	0.92
RG-D_N Fine-Tune	**0.79**	**0.93**	**0.93**	**96.09**	**0.93**

of this is twofold. Not only does this show that the model has learned what a building is on a general level, but also, exemplifies the significance of incorporating geometric data into the network.

Lastly, the weights from their respective models trained using the improved labels are used as pre-trained weights for fine-tuning on the Potsdam dataset. This causes significant improvements to all performance measures, with F1 scores ranging from 0.91 to 0.93. In all instances precision and recall were equal suggesting a stable model. Here the test dataset comprised of 20% of the total data. This is in contrast to the generic model tests as these could be tested against 100% of the dataset. Due to the size and homogeneous nature of the Potsdam dataset it is not possible to confidently confirm whether the model has simply over-fit to the data or indeed the solution is more general.

4.2. RetinaNet

Experiments undertaken with the RetinaNet architecture followed similar patterns to those observed with the Faster R-CNN architecture. However, the models trained using RetinaNet appear to have performed to a higher standard w.r.t the evaluation metrics on the main aerial dataset (Table 4). Models trained using public datasets obtain substantially higher mAP scores, with RGB, RG-D_E and RG-D_N achieving

Table 4
RetinaNet segmentation results. mAP is calculated for recall >0.5 with true positives defined as IoU >0.5.

Model	mAP	Precision	Recall	Accuracy (%)	F1 Value
RGB OS labels	0.53	0.84	0.78	95.12	0.81
RG-D_E OS labels	0.67	0.84	0.89	96.35	0.86
RG-D_N OS labels	0.67	0.86	0.92	96.82	0.89
RGB labels +	0.83	0.96	0.94	98.42	**0.94**
RG-D_E labels +	0.90	0.90	0.91	97.41	0.88
RG-D_N labels +	**0.92**	**0.95**	**0.95**	**98.62**	**0.94**

17%, 24% and 31% increases respectively over their Faster R-CNN counterparts.

Whilst the final per-pixel segmentation process differs between the two network pipelines many similarities are observed. In line with Mask R-CNN, the presence of a depth channel offered little improvement to precision, however, strongly influenced the recall values. Here, the recall values observed are 0.78, 0.89 and 0.92 for RGB, RG-D_E and RG-D_N respectively. This has resulted in a general overall performance increase in F1 score. The improvement does however come at a cost of computation time. Despite MorphGACs performing substantially faster than non-morphological GACs they are still comparatively slower than a typical CNN based inference. Whereas a single inference takes 0.5 s, a single image containing 10 buildings takes 30 s.[5] By re-implementing the MorphGAC to process on the GPU computation times could likely be largely decreased.

The use of the improved labels for model training, as with Mask R-CNN, had a definite impact on loss and accuracy during network training (Fig. 8). This resulted in mAP values to reach 0.83, 0.9 and 0.92 for RGB, RG-D_E and RG-D_N respectively. These results suggest that the model robustly detects buildings, with almost no buildings not being identified on the test set inference. This resulted in improved recall values for all scenarios. Such improvements suggest mostly false negatives have been resolved indicating less building have been missed, as apposed to false building detections being resolved. In contrast to Faster R-CNN, RetinaNet learns well when depth information isn't present as well as when it is. This is justified by the final RGB F1 value exceeding RG-D_E and equalling RG-D_N.

The increase performance in mAP over Faster R-CNN in the main building dataset was also present with the Potsdam dataset. This strengthens the evidence that the RetinaNet as an object detector had learned the characteristics of buildings more effectively (Table 5). This further indicates the potential benefits of the focal loss algorithm, and its relevance in single-class detection systems. Despite the advantages seen during object detection, over the Potsdam dataset the use of MorphGAC's as an online segmentation method was not as effective as Mask R-CNN. The most noticeable performance caveat is the recall scores for all scenarios. Whilst the high precision values (0.91–0.95) can be strongly accredited to the accurate initial seed points derived from RetinaNet, the poor recall would be accredited to the MorphGAC segmentation. More specifically, it is observed that the buildings are under-segmented. The Potsdam dataset contains a largely different building architecture and layout design to the majority of buildings in the main dataset. For example, much of the scene contains large continuously terraced buildings (Fig. 9). As this classifies as a single building it is therefore segmented in a single MorphGAC minimisation. However, the terrace buildings contain large depth variance across individual dwellings amongst the whole terrace. This therefore terminates the segmentation leaving the remainder of the building classed as background.

The online use of the MorphGAC segmentation is also the reason for little variance in F1 values across all of the scenarios. Generally, the performance of the building detector was high enough for all scenarios for the segmentation to be the main caveat. However, there was still a substantial performance increase in mAP for the fine tuned scenarios. The inclusion of depth data showed no significant improvements. Unfortunately, as the dataset is significantly smaller than what is recommended for training deep CNNs, it is a possibility the network has strongly over-fitted to the training data, and the lack of variance between the training and test produces seemingly very good results.

[5] On a single GPU for CNN inference and a 8 core CPU with multi-thread parrelisation for MorphGAC inference.

Table 5
Potsdam RetinaNet segmentation results. mAP is calculated for recall > 0.5 with true positives defined as IoU > 0.5.

Model	mAP	Precision	Recall	Accuracy (%)	F1 Value
RGB OSLabel Generic	0.46	0.91	0.72	93.94	0.80
RG-D_E OSLabel Generic	0.46	0.93	0.78	95.32	0.85
RG-D_N OSLabel Generic	0.48	0.94	0.79	95.54	0.86
RGB Label + Generic	0.50	0.93	0.75	92.91	0.83
RG-D_E Label + Generic	0.40	0.92	0.79	93.21	0.85
RG-D_N Label + Generic	0.48	0.93	0.78	94.87	0.85
RGB Fine-Tune	**0.81**	0.94	0.78	94.89	0.85
RG-D_E Fine-Tune	0.80	**0.95**	0.81	95.67	0.87
RG-D_N Fine-Tune	**0.81**	**0.95**	**0.82**	**96.12**	**0.88**

5. Discussion

The results presented in this paper reflect strongly the importance of improved labels for both object detection and segmentation. This was most evident for object detection, where the was an average of 47% increase in mAP score against 11.36% F1 segmentation score. It was observed that improving labels had the largest impact on the precision values for segmentation. Despite the fact over-generalisation would effect the mean depth channel value the greatest, the largest increases for both object detection and segmentation were observed in the RGB scenarios. This indicates the relevance the results from this study have on standard RGB aerial datasets. Furthermore, this suggests that poor labelling overlapping incorrect textures (i.e. roof vs road/garden) is potentially more detrimental to the networks performance than a constant mean difference. This is not surprising as CNNs are known to learn textures in many of the intermediate layers. These results offer quantitative analysis of the hypothesis that large-scale GIS data can be used to train networks, with sheer quantity counteracting poor quality. We demonstrate here that valuable results can be obtained with low quality training data, however, it is essential to have high quality labels for state-of-the-art results. Poor quality can be a consequence of a number of causes such as; over-generalisation, data acquisition temporal variance, missing data, etc. however, when using depth data, the quality of aerial image orthorectification is prominent. This was witnessed extensively across both datasets, where errors in othorectification led to a small misalignment in the aerial image and depth data.

The MorphGACs proved to be an effective tool for pixel-wise segmentation of buildings from initial building footprints. However, the tool was not as effective when continuous terrace buildings as seen in the Potsdam dataset, dominate the local architecture. This demonstrates one of the fundamental weaknesses of rule-based algorithms. When faced with new, unexpected data, they often perform poorly. This fact is perhaps one of the largest driving factors for machine learning research. Despite this, we demonstrate here that rule-based algorithms can still act as a valuable tool to aid the training of machine learning models. Furthermore, MorphGAC derived segmentation on the UK dataset was showing to achieve higher accuracy than the state-of-the-art Mask R-CNN for segmentation, provided seed points were given. This was demonstrated by the success of the RetinaNet segmentation compared to Mask R-CNN (0.82 and 0.92 F1 scores respectively). The results here demonstrate that whilst a fully end-to-end deep learning approach would be more desirable due to increased robustness and computation speed, in many specific applications a combination of bespoke rule-based and deep learning approaches can yield the greatest performance.

The potential benefits of the inclusion of depth data in aerial object detection and segmentation scenarios is also exemplified from our experiments. We demonstrate that existing CNN architectures (RetinaNet and Faster/Mask R-CNN) can be used without alteration, and therefore, bespoke networks to handle depth information are not essential. This was most prominent in the large UK building datasets where the

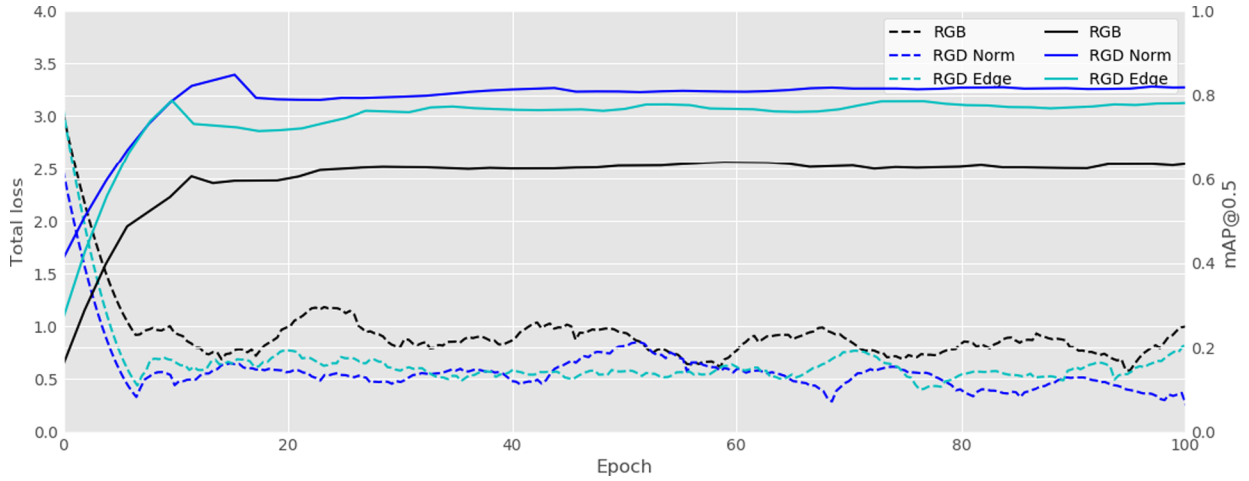

Fig. 6. Training plots for datasets using the improved labels. Models containing depth information out perform the standard RGB dataset. RG-DRG-D_N converges with the lowest loss value and highest mAP score.

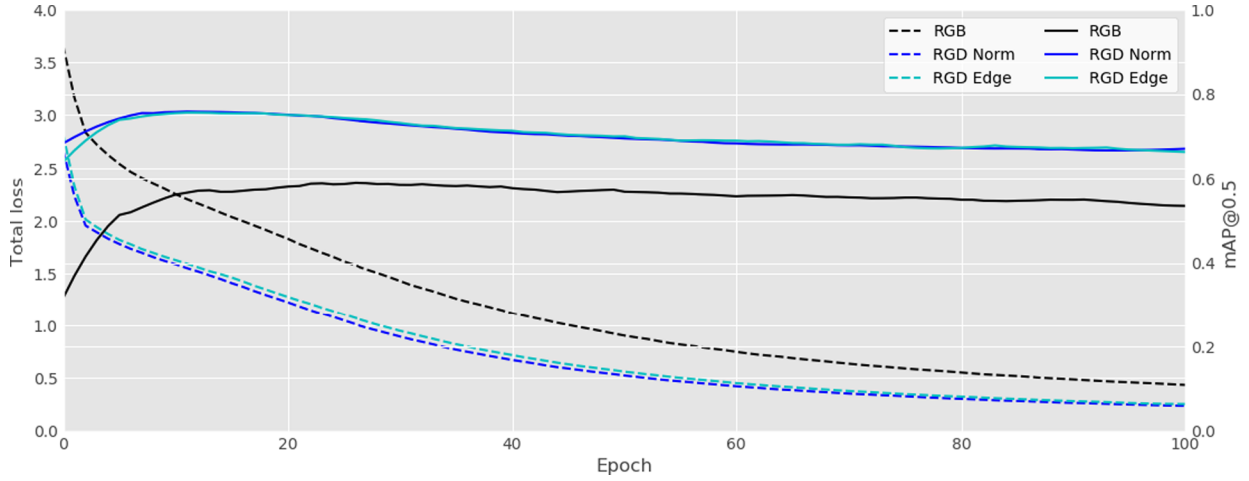

Fig. 7. RetinaNet training plots for public ground truth label datasets. The inclusion of depth data results in both lower loss convergence values and high mAP scores. There is no immediate discrepancies between the use of edge magnitude (RG-D_E) and normalised (RG-D_N) depth channels.

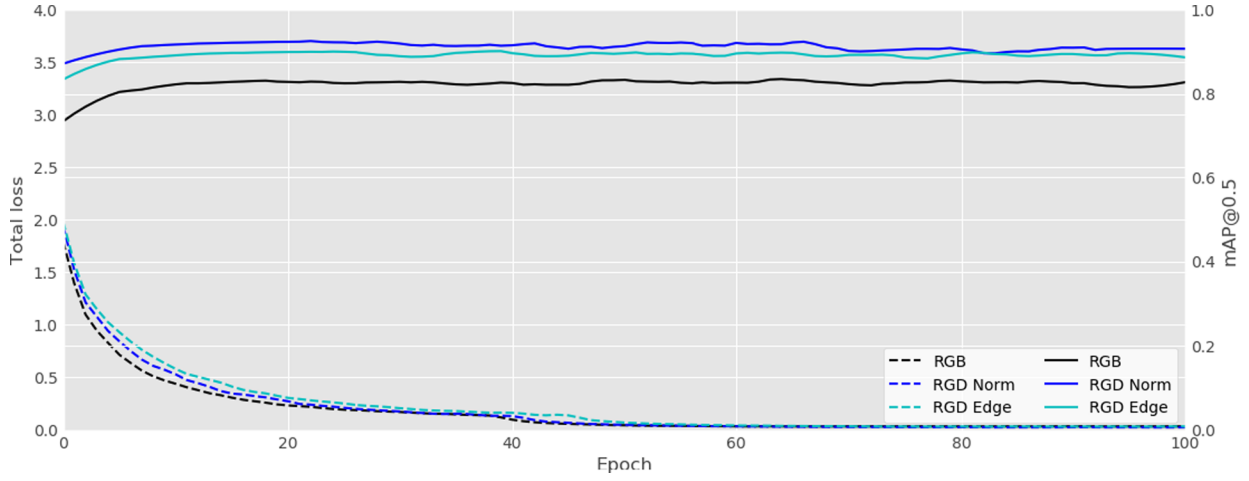

Fig. 8. RetinaNet training plots for datasets with improved ground truth labels. All scenarios demonstrate higher mAP values compared to their respective datasets using public data ground truth labels. This is also represented by substantially lower and earlier loss convergence values.

inclusion of depth data resulted an overall increased mAP for object detection and improved recall values for Mask R-CNN segmentation. Despite this the implications of the depth channel were not as clear when evaluating on the Potsdam benchmark dataset. In all scenarios network inferences after fine-tuning on the Potsdam dataset very high performance results were achieved. It could be assumed that the larger dataset is more indicative of the general model performance and therefore this suggests the high results observed on the fine-tuned

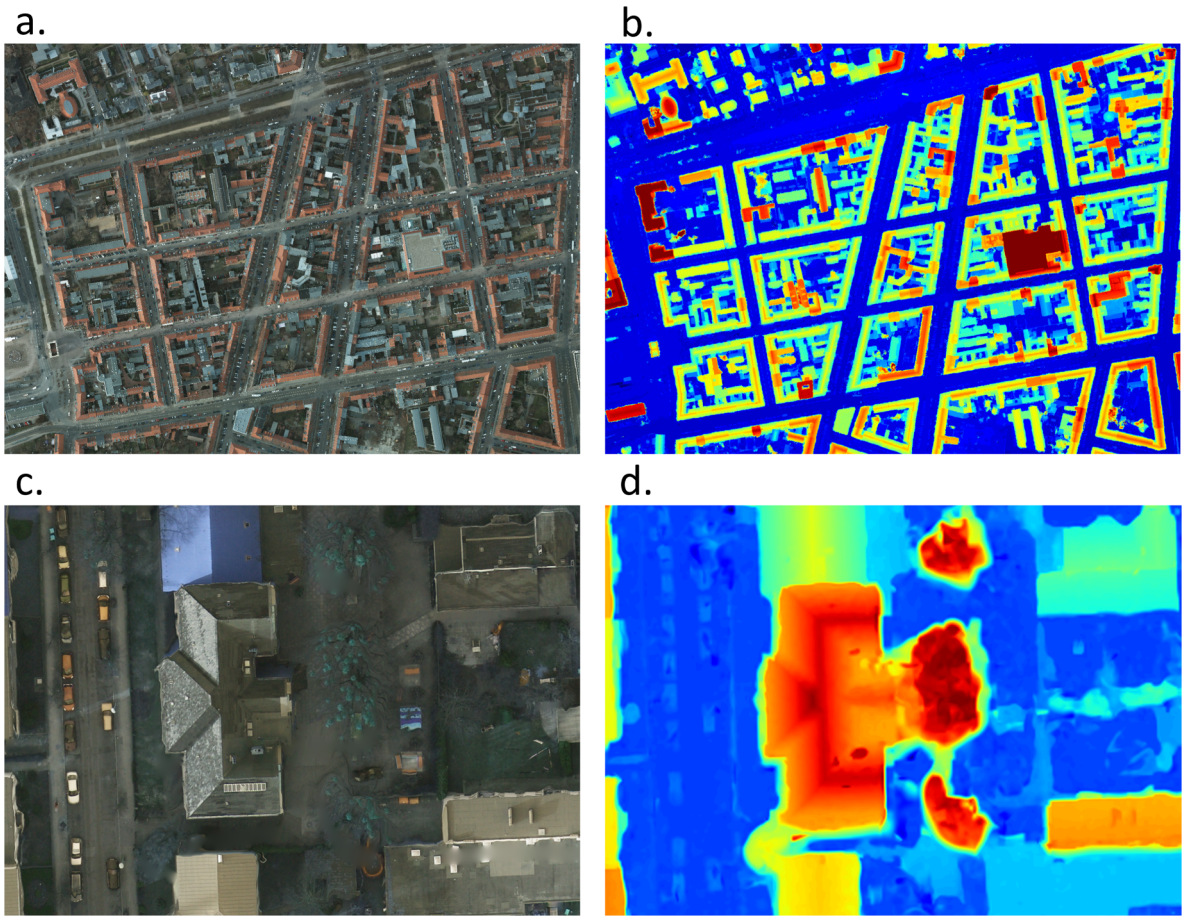

Fig. 9. (a) Aerial and DSM of Potsdam town design. Many of the buildings are continuous terrace houses, which can cause issues for a MorphGAC segmentation. (b) Example of apparenty DSM smeering. In the aerial image the tree appears to be seperated from the house, however, the DSM shows a definitive overlap.

datasets are caused by a form of over-fitting. The Potsdam benchmark, when tiled to a similar geographic size ($250m^2$) to the UK dataset only consisted of 133 images of size (600x600px). All network architectures had >50 million parameters, and therefore, it is unreasonable to conclude exactly what the model has learned, and how applicable the models would be for applications in different datasets. This highlights an issue where relatively small benchmark datasets are inferred with increasingly deep neural network architectures. Furthermore, the network weights pre-trained on the UK dataset also did not perform well on the Potsdam benchmark. This indicates that even though a large dataset was used for training, the general features of UK buildings were not enough to allow for reliable inference on substantially different building/street designs. This is important as this demonstrates that features learned from the depth channel alone is not robust enough for a functional general model. Instead, higher layer features which are associated with larger patches of the image and therefore object context are required within the training dataset. Despite this, the data aquisition methodology presented here is scalable and therefore incorporation of more diverse training data would be easily achieved and likely highly beneficial.

The results presented in this paper suggest that RG-D_N is the most effective dataset to build on, exaggerating building edges of the depth channel by computing a differential demonstrated no immediate advantages. On the contrary, the RG-D_E consistently performed (albeit very slightly) lower than the RG-D_N datasets. Moreover, the network tended to learn slower with the training loss converging later than RG-D_N (Figs. 5–8). This highlights that even though the pre-trained weights from ImageNet included blue channels instead of depth, some of this information was transferable to the normalised depth channel. Further illustrating the potential gains from using pre-trained weights, as even if higher layer features are drastically different to the desired weights, lower level features remain relevant (Fig. 10).

The comparison between the two-stage Faster R-CNN and one-stage RetinaNet, concluded strongly in favour of RetinaNet. This is quantified by an average increase in mAP of 1.25 on the main UK building dataset and 0.078 overall. This equates to a percentage increase of 20.13% and 17.79% respectively. An increase was observed for all but one processing scenario when using RetinaNet. This consistency implies models trained using the RetinaNet architecture both more reliable and robust. Moreover, this performance increase comes at a computation speed gain with the average inference speed amounting to 73 ms and 89 ms for RetinaNet and Faster R-CNN respectively. Although this contradicts the classic speed/accuracy trade-off (Huang et al., 2017), this instead demonstrates the potential benefits of the focal loss algorithm introduced in RetinaNet. In the UK building dataset the absolute pixel class imbalance observed was 1–6.66 and 1–4.5 for the Potsdam dataset, for foreground (building) and background respectively. This however becomes exemplified by classification at multiple scales. Therefore, this suggests that for single-class or multi-class where large class imbalances are still present the focal loss is of fundamental use for high performing models. Such conclusions were also made in a previous study (Griffiths and Boehm, 2018). Moreover, Lin et al. (2017) demonstrate that even in the COCO benchmark dataset where class imbalances are not as prominent, the model offers performance improvements. The results indicate that the Mask R-CNN could benefit from the incorporation of a focal loss method into the object detection branch. This would combine the high accuracy object detection of the RetinaNet with power of semantic segmentation.

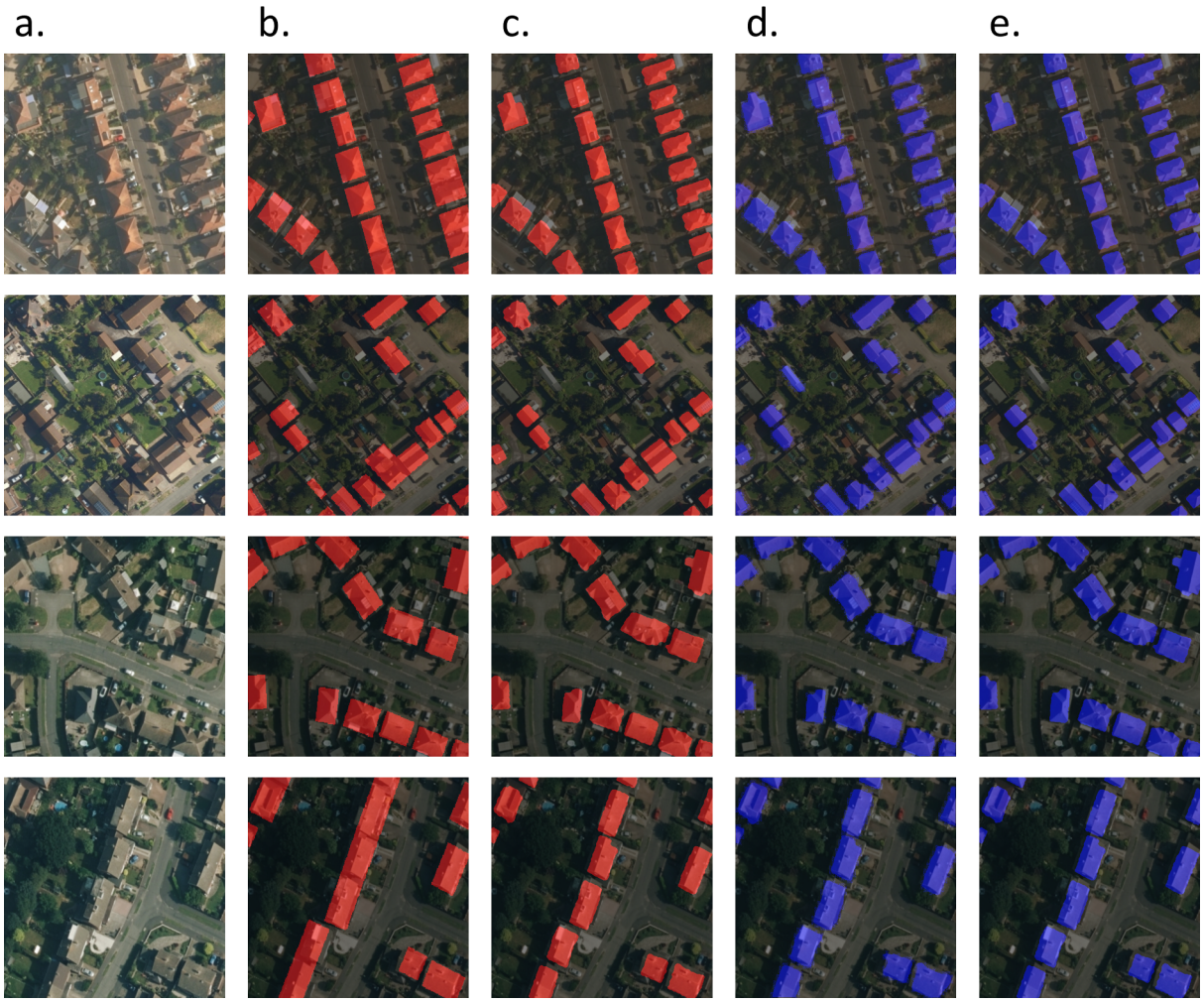

Fig. 10. (a) Aerial images of UK buildings to be segmented. (b) Public GIS data labels (c) Improved labels using the active contours (d) Mask R-CNN segmentation results (e) RetinaNet object detection + active contour segmentation results.

6. Conclusion

In this paper we quantify the benefit of using improved publicly available ground truth labels for building segmentation through empirical means. Three CNN architectures were subsequently trained; Faster R-CNN, RetinaNet and Mask R-CNN. MorphGACs were then utilised on the image depth channel to improve the quality of the labels. This was achieved using course low-resolution labels as initial seed points, determining label over-generalisation and finally expanding until settling on the sharp geometric boundaries of buildings. Despite robustness and competency of segmentation in the UK dataset, for which the algorithm was designed, this did not work universally as demonstrated on the Potsdam dataset. This demonstrates the potential use for bespoke rule-based algorithms to aid the training of CNNs, however, for online use demonstrates the advantages machine learning based approaches offer in terms of robustness in new scenarios. Improvements were observed using the improved labels on RGB, RG-D_E and RG-D_N datasets for all network architectures. The most significant being in the Faster R-CNN and Mask R-CNN architectures where an average improvement of 42% in mAP score and 8% in F1 segmentation score were observed respectively. The results suggest that substantial performance gains can be made for both RGB and RG-D datasets by improving label quality, even when trained over a large dataset. Models trained with the RetinaNet architecture demonstrated an average mAP score of 1.25 over the Faster R-CNN. This is likely accredited to the use of the focal loss algorithm. Further indicating the addition of a focal loss method into the Faster R-CNN branch of the Mask R-CNN would result in improved results.

In line with previous studies, the addition of lidar-derived 2.5-D depth data aided the training of CNNs for building segmentation. This was achieved by replacing the blue channel of 25,556 aerial images containing 169,835 instances of buildings. The results demonstrated a significant improvement in model performance when depth data was present. Most noticeable was the correction of false negatives in the image segmentation, which resulted in a large increase in evaluated recall values. Two different depth channel inputs were tested. This first RG-D_E was created by taking a differential of the image to determine the images edge-magnitude, which exaggerated sharp changes in depth and therefore building edges. The second, RG-D_N was created by normalising each image tile between 0 and 255 to allow the network to learn realitive values within the local context of the buildings surroundings. The results concluded that RG-D_E tended to learn slower, however, converged on similar accuracy and loss values to the RG-D_N training examples. Despite this, RG-D_N on average performed marginally better on all performance measures. Therefore, we conclude this is the most suitable depth channel formation.

Acknowledgements

The work carried out in this paper is partially funded by Bentley Systems.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Software available from tensorflow.org.

Aijazi, A.K., Checchin, P., Trassoudaine, L., 2013. Segmentation based classification of 3D urban point clouds: a super-voxel based approach with evaluation. Remote Sens. 5 (4), 1624–1650.

Akel, N.A., Zilberstein, O., Doytsher, Y., 2004. A Robust Method Used With Orthogonal Polynomials and Road Network For Automatic Terrain Surface Extraction From Lidar Data In Urban Areas, 34, 6.

Álvarez, L., Baumela, L., Henríquez, P., Márquez-Neila, P., 2010. Morphological snakes. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2197–2202.

Awrangjeb, M., Ravanbakhsh, M., Fraser, C.S., 2010. Automatic detection of residential buildings using LIDAR data and multispectral imagery. ISPRS J. Photogramm. Remote Sens. 65 (5), 457–467.

Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39 (12), 2481–2495.

Bittner, K., Adam, F., Cui, S., Körner, M., Reinartz, P., 2018. Building footprint extraction from vhr remote sensing images combined with normalized dsms using fused fully convolutional networks. IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens. 11 (8), 2615–2629.

Caselles, V., Kimmel, R., Sapiro, G., 1997. Geodesic active contours. Int. J. Comput. Vision 22 (1), 61–79.

Chan, T.F., Vese, L.A., 2001. Active contours without edges. IEEE Trans. Image Process. 10 (2), 266–277.

Chen, D., Zhang, L., Mathiopoulos, P.T., Huang, X., 2014. A methodology for automated segmentation and reconstruction of urban 3-D buildings from ALS point clouds. IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens. 7 (10), 4199–4217.

Chen, K., Weinmann, M., Sun, X., Yan, M., Hinz, S., Jutzi, B., Weinmann, M., 2018. Semantic segmentation of aerial imagery via multi-scale shuffling convolutional neural networks with deep supervision. ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. 4 (1).

Chen, L., Wang, S., Fan, W., Sun, J., Naoi, S., 2015. Beyond human recognition: a CNN-based framework for handwritten character recognition. In: Pattern Recognition (ACPR), 2015 3rd IAPR Asian Conference On. IEEE, pp. 695–699.

Ciresan, D., Giusti, A., Gambardella, L.M., Schmidhuber, J., 2012. Deep neural networks segment neuronal membranes in electron microscopy images. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems 25. Curran Associates, Inc., pp. 2843–2851.

Cohen, L.D., 1991. On active contour models and balloons. CVGIP: Image Understand. 53 (2), 211–218.

Delassus, R., Giot, R., 2018. Cnns fusion for building detection in aerial images for the building detection challenge. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, pp. 237–2374.

Dorninger, P., Pfeifer, N., 2008. A comprehensive automated 3D approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds. Sensors 8 (11), 7323–7343.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR. IEEE, pp. 580–587.

Griffiths, D., Boehm, J., 2018. Rapid object detection systems, utilising deep learning and unamanned aerial systems (UAS) for civil engineering applications. ISPRS – Int. Arch. Photogram. Remote Sens. Spatial Inf. Sci. XLII-2, 391–398.

Guo, L., Chehata, N., Mallet, C., Boukir, S., 2011. Relevance of airborne lidar and multispectral image data for urban scene classification using Random Forests. ISPRS J. Photogramm. Remote Sens. 66 (1), 56–66.

Guo, Z., Shao, X., Xu, Y., Miyazaki, H., Ohira, W., Shibasaki, R., 2016. Identification of village building via google Earth images and supervised machine learning methods. Remote Sens. 8 (4), 271.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask R-CNN. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2980–2988.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1026–1034.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. IEEE 770–778.

Hernandez, J., Marcotegui, B., 2009. Point cloud segmentation towards urban ground modeling. In: 2009 Joint Urban Remote Sensing Event, pp. 1–5.

Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., Murphy, K., 2017. Speed/accuracy trade-offs for modern convolutional object detectors. IEEE 3296–3297.

Huang, J., Zhang, X., Xin, Q., Sun, Y., Zhang, P., 2019. Automatic building extraction from high-resolution aerial images and lidar data using gated residual refinement network. ISPRS J. Photogramm. Remote Sens. 151, 91–105.

Huertas, A., Nevatia, R., 1988. Detecting buildings in aerial images. Comput. Vision Graph. Image Process. 41 (2), 131–152.

Ioffe, S., Szegedy, C., 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. CoRR abs/1502.03167.

Kampffmeyer, M., Salberg, A.-B., Jenssen, R., 2016. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2016 IEEE Conference On. IEEE, pp. 680–688.

Kanopoulos, N., Vasanthavada, N., Baker, R.L., 1988. Design of an image edge detection filter using the Sobel operator. IEEE J. Solid-State Circ. 23 (2), 358–367.

Kass, M., Witkin, A., Terzopoulos, D., 1988. Snakes: active contour models. Int. J. Comput. Vision 1 (4), 321–331.

Kraus, K., Pfeifer, N., 2001. Advanced DTM generation from LIDAR data. In: Proceedings of the ISPRS Workshop on Land Surface Mapping and Characterization Using Laser Altimetry.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105.

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D., 1989. Backpropagation applied to handwritten zip code recognition. Neural Comput. 1 (4), 541–551.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2324.

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017. Feature pyramid networks for object detection. In: CVPR, vol. 1. pp. 4.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. 'Focal loss for dense object detection', arXiv preprint arXiv: 1708.02002.

Liow, Y.-T., Pavlidis, T., 1990. Use of shadows for extracting buildings in aerial images. Comput. Vision Graph. Image Process. 49 (2), 242–277.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C., 2016. Ssd: single shot multibox detector. In: European Conference on Computer Vision. Springer, pp. 21–37.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', pp. 3431–3440.

Marmanis, D., Schindler, K., Wegner, J.D., Galliani, S., Datcu, M., Stilla, U., 2018. Classification with an edge: improving semantic image segmentation with boundary detection. ISPRS J. Photogramm. Remote Sens. 135, 158–172.

Marmanis, D., Wegner, J.D., Galliani, S., Schindler, K., Datcu, M., Stilla, U., 2016. Semantic segmentation of aerial images with an ensemble of CNNs. ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. 3, 473.

Márquez-Neila, P., Baumela, L., Alvarez, L., 2014. A morphological approach to curvature-based evolution of curves and surfaces. IEEE Trans. Pattern Anal. Mach. Intell. 36 (1), 2–17.

Paisitkriangkrai, S., Sherrah, J., Janney, P., Van-Den Hengel, A., 2015. Effective semantic pixel labelling with convolutional networks and Conditional Random Fields. IEEE 36–43.

Quang, N.T., Thuy, N.T., Sang, D.V., Binh, H.T.T., 2015. An efficient framework for pixel-wise building segmentation from aerial images. In: Proceedings of the Sixth International Symposium on Information and Communication Technology. ACM, pp. 282–287.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster R-CNN: towards real-time object detection with region proposal networks. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 28. Curran Associates Inc, pp. 91–99.

Saeedi, P., Zwick, H., 2008. Automatic building detection in aerial and satellite images. In: Robotics and Vision 2008 10th International Conference on Control, Automation, pp. 623–629.

Saito, S., Aoki, Y., 2015. Building and road detection from large aerial imagery. In: Image Processing: Machine Vision Applications VIII', vol. 9405 International Society for Optics and Photonics.

Secord, J., Zakhor, A., 2007. Tree detection in urban regions using aerial lidar and image data. IEEE Geosci. Remote Sens. Lett. 4 (2), 196–200.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y., 2013. Overfeat: integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv: 1312.6229.

Simonyan, K., Zisserman, A., 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition, arXiv: 1409.1556 [cs].

Sirmacek, B., Unsalan, C., 2008. Building detection from aerial images using invariant color features and shadow information. IEEE 1–5.

Socher, R., Huval, B., Bath, B., Manning, C.D., Ng, A.Y., 2012. Convolutional-recursive deep learning for 3d object classification. In: Advances in Neural Information Processing Systems, pp. 656–664.

Strom, J., Richardson, A., Olson, E., 2010. Graph-based segmentation for colored 3D laser point clouds. In: Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference On. IEEE, pp. 2131–2136.

Sun, W., Wang, R., 2018. Fully convolutional networks for semantic segmentation of very high resolution remotely sensed images combined with dsm. IEEE Geosci. Remote Sens. Lett. 15 (3), 474–478.

Suzuki, S., Be, K., 1985. Topological structural analysis of digitized binary images by border following. Comput. Vision Graph. Image Process. 30 (1), 32–46.

Vakalopoulou, M., Karantzalos, K., Komodakis, N., Paragios, N., 2015. Building detection in very high resolution multispectral data with deep learning features. In: 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 1873–1876.

Vetrivel, A., Gerke, M., Kerle, N., Vosselman, G., 2015. Segmentation of UAV-based images incorporating 3D point cloud information. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. 40 (3), 261.

Vosselman, G., 2000. Slope based filtering of laser altimertry data. Int. Arch. Photogramm. Remote Sens. 935–942.
Wu, G., Shao, X., Guo, Z., Chen, Q., Yuan, W., Shi, X., Xu, Y., Shibasaki, R., 2018. Automatic building segmentation of aerial imagery using multi-constraint fully convolutional networks. Remote Sens. 10 (3), 407.
Xia, G.-S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M., Zhang, L., 2018. DOTA: A Large-scale Dataset for Object Detection in Aerial Images. In: IEEE CVPR.
Xiao, J., Zhang, J., Adler, B., Zhang, H., Zhang, J., 2013. Three-dimensional point cloud plane segmentation in both structured and unstructured environments. Robot. Auton. Syst. 61 (12), 1641–1652.
Yuan, J., 2016. 'Automatic building extraction in aerial scenes using convolutional networks. arXiv preprint arXiv: 1602.06564.
Zhang, Q., Wang, Y., Liu, Q., Liu, X., Wang, W., 2016. CNN based suburban building detection using monocular high resolution Google Earth images. In: 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 661–664.