International Journal of Geo-Information

MDPI

*Article*

# Simultaneous Extraction of Road and Centerline from Aerial Images Using a Deep Convolutional Neural Network

**Tamara Alshaikhli *, Wen Liu and Yoshihisa Maruyama**

Graduate School of Engineering, Chiba University, Chiba 263-8522, Japan; wen.liu@chiba-u.jp (W.L.); ymaruyam@tu.chiba-u.ac.jp (Y.M.)

* Correspondence: tamara_alshaikhli@chiba-u.jp; Tel.: +81-43-290-3555

**Abstract:** The extraction of roads and centerlines from aerial imagery is considered an important topic because it contributes to different fields, such as urban planning, transportation engineering, and disaster mitigation. Many researchers have studied this topic as a two-separated task that affects the quality of extracted roads and centerlines because of the correlation between these two tasks. Accurate road extraction enhances accurate centerline extraction if these two tasks are processed simultaneously. This study proposes a multitask learning scheme using a gated deep convolutional neural network (DCNN) to extract roads and centerlines simultaneously. The DCNN is composed of one encoder and two decoders implemented on the U-Net backbone. The decoders are assigned to extract roads and centerlines from low-resolution feature maps. Before extraction, the images are processed within an encoder to extract the spatial information from a complex, high-resolution image. The encoder consists of the residual blocks (Res-Block) connected to a bridge represented by a Res-Block, and the bridge connects the two identical decoders, which consists of stacking convolutional layers (Conv.layer). Attention gates (AGs) are added to our model to enhance the selection process for the true pixels that represent road or centerline classes. Our model is trained on a dataset of high-resolution aerial images, which is open to the public. The model succeeds in efficiently extracting roads and centerlines compared with other multitask learning models.

**Keywords:** multitask learning; deep convolutional neural network; attention gates; aerial images; extraction of road and centerline; simultaneous extraction process; residual blocks



**Citation:** Alshaikhli, T.; Liu, W.; Maruyama, Y. Simultaneous Extraction of Road and Centerline from Aerial Images Using a Deep Convolutional Neural Network. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 147. https://doi.org/10.3390/ijgi10030147

Academic Editors: Wolfgang Kainz and Giuseppe Borruso

Received: 26 January 2021
Accepted: 7 March 2021
Published: 8 March 2021



## 1. Introduction

Road extraction from remote sensing imagery is of considerable importance because it affects modern life specifically, such as city and urban planning [1], transportation engineering [2], and the operation of unmanned vehicles [3]. Therefore, this topic attracts researcher attention in attempting to address the challenges that affect the quality of extracted roads, and it consists of two correlated subtasks: road detection and centerline extraction [4]. Many researchers have studied this topic separately. For road extraction, their studies employed a hierarchical approach [5], a probabilistic approach [6], and deep learning algorithms [7–10]. These researchers studied pixel classification to classify road (foreground) or nonroad (background) pixels.

Centerline extraction extracts the pixels located at the center of the road. Generally, it is very difficult to extract centerlines from aerial images directly. The centerline needs to be extracted from road segmentation [11] using several algorithms, such as morphological thinning algorithms [12,13]. Although the thinning algorithm is easy to implement, it has a drawback that affects the accuracy. The spurs are generated surrounding the centerline extracted by the thinning algorithm. The process of centerline extraction, as mentioned before, emphasizes the strong correlation between these two tasks.

The new evolution in deep learning helped to overcome many difficulties in different fields of science. The first convolutional neural network (AlexNet) by Krizhevsky and

Hinton [14] with five convolution layers and three fully connected layers paved the way for developing improved neural network architectures. Long et al. [15] presented a fully convolutional neural network (FCN), where they replaced all the fully connected layers with convolution layers, increased the efficiency of the predicted images. They implemented the FCN on the backbone of AlexNet [14], VGGNet [16], and GoogLeNet [17]. The FCN has been adapted by many neural network architectures. The backbone of the encoder and decoder helps to create U-Net [18], which is considered one of the best architectures for semantic segmentation using a small training dataset.

Training two subtasks separately may lead to the loss of considerable information while moving from one task to another. This was observed in previous researches where road and centerline extraction was performed separately. The concept of multitask learning (MTL) helps overcome this issue. According to Caruana [19], "it may be easier to learn several hard tasks at one time than to learn these same tasks separately." Although MTL is not a new concept, it is recently used for road and centerline extraction. The first attempt to train these two subtasks simultaneously was presented by Cheng et al. [4]. Their attempt consisted of two models and each model was assigned a task. In their methodology, the last deconvolution layer was shared from the road model as an input to the centerline model. However, their model was inadequate to predict a smooth centerline. To obtain the result for their centerline segmentation, they employed thinning algorithm on their network segmentation results. This resulted in the loss of the shared information between the two subtasks during the training process of their proposed model.

In this study, we propose our gated deep convolutional neural network (DCNN), which is implemented on the U-Net backbone with a combination of residual blocks (Res-Blocks) [20,21] in the encoder part. The two tasks share the same encoder, and the two decoders are assigned to perform each task: road and centerline extractions. The authors added attention gates (AGs) [22] for the better selection of pixels and more efficient prediction. Our DCNN is trained to extract roads and centerlines simultaneously. The architecture of our model shares the information of the road and centerline data in the training process without losing the extracted information.

The main contributions of this study are:

- It proposed a multitask learning (MTL) model for efficient extraction of roads and centerlines.
- It revealed the importance of MTL to improve the results of both tasks compared with that of our previous work.
- It proposed an appropriate MTL architecture to successfully train both the tasks.

The paper is organized as follows: Section 2 presents the related works on both road and centerline extraction tasks; Section 3 shows the proposed model and methodology; Section 4 implements the proposed models and evaluates the accuracy of the results; Section 5 summarizes the results and discussion of this study, and the conclusions are mentioned in Section 6.

## 2. Related Works

### *2.1. Road Extraction Method*

The road extraction task has attracted considerable attention, and different studies use different methods and techniques to extract roads from aerial images. Many early studies used classification-based methods according to the features, textures, and geometric features of roads [2]. Baumgartner et al. [23] proposed a model to extract roads from aerial imagery, which consisted of two parts. The first part focuses on the road characteristics (e.g., width and type). The second part assigns the local context of roads (e.g., background, buildings, and trees) to the global context. They found that the global context plays a major role in the quality of the results. Trinder and Wan [24] proposed a knowledge-based method based on Marr's theory of vision to extract roads from aerial images. Their methodology includes three types of processing: Low-level processing to extract features, midlevel processing for grouping and generating features, and high-level processing for

road recognition. Dal Poz et al. [25] presented an automatic method to extract roads from medium- and high-resolution aerial images. Their method consists of two consecutive steps. The first is the extraction of road seeds using a number of algorithms, such as the Canny edge detection method and edge linking algorithm, followed by split and merge algorithms. The second is road completion by linking the extracted seeds. Song and Civco [26] presented a pixel-object approach based on both classification and segmentation methods. In the classification phase, they used a support vector machine to extract road and nonroad pixels. In the segmentation phase, they used the region growing technique based on the similarity criterion to segment the whole road region. Previous works have focused on multiple stages based on several techniques and algorithms to extract roads from aerial images. Advanced studies have focused on machine learning to enhance the quality of extracted roads. Mnih and Hinton [27] applied a patch-based approach using a neural network with postprocessing. The network shows promising results compared with the previous techniques.

With the new evolution in deep learning by the first convolutional neural network (CNN) presented by Krizhevsky and Hinton [14], many studies have adopted the CNN concept, especially in road detection from aerial images. Saito et al. [28] extracted roads and buildings from aerial images using CNN with a patch-based approach similar to Mnih and Hinton [27]. Bastani et al. [29] presented a CNN-based search algorithm, and Mattyus et al. [30] proposed an algorithm to address the incomplete connection of roads from CNN output segmentation. These previous works represent the adaptation of the CNN concept. With the concept of the FCN by Long et al. [15], different methods and techniques were proposed to extract roads from aerial images. Zhang et al. [31] used the FCN with an ensemble strategy using different weights for the loss function. The existence of stronger architectures, such as U-Net [18], shows great improvement in the classification and segmentation process. Zhang et al. [7] used Res-U-Net, which consists of Res-Block implemented on the U-Net backbone, and their model showed a good result compared with different CNN architectures. Buslaev et al. [8] proposed a model whose encoder part consists of ResNet-34 [20], while the decoder is implemented on a vanilla U-Net. Previous works that adopted U-Net architectures show promising results, specifically in the remote sensing field.

### *2.2. Centerline Extraction Method*

Generally, extraction of centerlines directly from aerial imagery is considered a very difficult task. Reviewing the previous works, centerline extraction is mainly performed in two steps: Extraction of road segments and extraction of centerline pixels from road segments. Due to the relationship between the two tasks, the centerline extraction went through the same evolution as the road extraction. Guo et al. [32] extracted centerlines following multistage processes. First, roads were extracted from low- and moderate-resolution aerial images based on a detection algorithm. Second, the centerline is estimated by thinning the road segment based on the Newton and square methods. Shi et al. [12] extracted centerlines from main urban roads following four steps. First, they extract roads by purposing spectral-spatial classification to classify aerial images into two groups: roads and others. The second step focuses on improving the extracted road. If the roads are located in the homogeneous region, they are emphasized by local Geary's C. The third step defines the final road segmentation with shape feature filtering. The last step is to extract the centerline by using local linear kernel regression. Sujatha and Selvathi [33] extracted centerlines from high-resolution satellite imagery in three steps. First, road segments are extracted using histogram analysis, applying adaptive global thresholds to select the road pixels. Then, the connected components are extracted by dilation and mathematical intersection combinations, such as closing morphological approaches, which closes the holes in the road component and clears the image from unwanted portions. Finally, the morphological thinning algorithm is applied to extract the centerline.

Previous works focused on extracting a clear and connected road first. In the final step, the centerline was extracted using different algorithms. Hence, the result of centerline extraction is strongly dependent on road extraction, and it is difficult to extract the centerline directly from aerial imagery. Recently, with advanced research in deep learning, the MTL has been introduced to train different multitask problems. Shen et al. [34] presented an MTL-CNN to identify object skeletons in natural images. The first task is to define whether the pixel belongs to the skeleton by skeleton localization, and the second task is to predict the skeleton scale. Xiao et al. [35] also presented a skeleton extraction model from natural images, and they employed the multi-class imbalanced dataset. Liu et al. [36] investigated the problem of shared private spaces of the latent feature on text, via adversarial MTL to enhance the text sequence on 16 different text classifications. Chen et al. [37] presented a model based on MTL-CNN for gadolinium-enhanced magnetic resonance images (GE-MRI) images to accomplish two tasks: atrial segmentation and classification of pre/postablation. Gonçalves et al. [38] performed segmentation and recognition of license plates using both high- and low-resolution images by MTL-CNN.

The aforementioned MTL concept has been used for different tasks in various fields. However, the extraction of road and centerline simultaneously based on the MTL concept has not been extensively studied. The first research work was presented by Cheng et al. [4]. They proposed a cascaded DCNN that consists of two models. One is to extract roads, and the other is for centerline extraction. The two models are concatenated with the last deconvolution layer of the road extraction model, which is used as an input for the centerline extraction model. Although the model showed promising results, they still employed a morphological thinning algorithm for the centerline segmentation results to enhance the prediction. The MTL scheme used in their model suffers, due to the weak sharing of information between two subtasks. Yang et al. [39] presented a recurrent CNN based on the U-Net backbone (RCNN-U-Net). They proposed a full MTL to extract roads and centerlines simultaneously from aerial images, and their work showed better results compared with that of their previous approaches. Liu et al. [40] proposed an MTL neural network named RoadNet to perform simultaneous road and centerline extraction, and it showed promising results. However, the results for centerline extraction were inadequate, especially at the intersection. Recently, Shao et al. [41] proposed an MTL model with pyramid scene parsing (PSP) pooling to extract road and centerline from very high-resolution aerial images in urban areas. Their research is promising. However, the results obtained in their study require improvement. The MTL proposed in their study achieved an intersection over union (IOU) of 0.5553 and F1-Score of 0.7141 for road extraction, and an IOU of 0.5395 and F1-Score of 0.7009 for centerline extraction.

As mentioned above, road extraction performed by early studies involved processes within several stages and steps using different methods and algorithms. The results had several problems, such as disconnection of the roads and difficulties in extracting a clear segmentation for the roads due to the complex image background: shadows of buildings and trees. Additionally, for centerline extraction, they had to extract the road segments first. Using multistage methods, the centerlines were finally extracted. Due to the multistage method in the extraction process, the results were not perfect for both classes: road and centerline. Advanced research in deep learning and the adoption of the MTL concept enables the simultaneous sharing of information between the two classes in the training and extraction processes, which leads to enhanced results. In this study, we adopt the MTL concept by proposing our model to address the extraction task for roads and centerlines simultaneously.

## 3. Proposed Model

This study introduces an extension of our first work [42]. The work was proposed to extract only roads from aerial images. In this study, we extended our proposed method to extract roads and centerlines simultaneously based on multitask learning.

In this section, we explain the methodology of this research. First, the basic backbone of our architecture is explained. Subsequently, detailed descriptions of the four proposed models are presented—highlighting their similarities and differences.

### 3.1. Basic Backbone of Architecture

To explain our proposed model, we first explain the basic backbone of our first model [42] based on single task learning (STL). The model is applied to the U-Net [18] architecture, which consists of contracting and expanding parts. We call them encoding and decoding parts, respectively. In the encoding part, there are three residual blocks (Res-Block) [20,21]. Each Res-Block consists of two convolutional layers (Conv.layers). The first one has a stride of 2 to downsize the feature map before the next Conv.layer. Before each Conv.layer, batch normalization (BN) is performed, followed by an activation function (ReLU). The number of filters for each Res-Block is 64, 128, and 256. In the decoding part, there are also three blocks, and each block consists of two Conv.layers with 265, 128, and 64 filters. Before the Conv.layer, there is a concatenate layer that connects each block in the decoder with the corresponding Res-Block in the encoder through the skip connection of the U-Net. The concatenate layer is implemented after an upsampling layer with a stride of 2. There is one Res-Block between the encoding and decoding parts with the 512 filters. The feature map of the last Conv.layer in the decoding part is processed in two additional Conv.layers after upsampling the last feature map. All the layers have a filter size of $3 \times 3$. The last layer has a $1 \times 1$ filter size, where a sigmoid activation function is implemented to produce a binary output for road and nonroad classes. It should be noted that the input image size is $224 \times 224$, similar to ResNet [20,21]. Figure 1 represents the basic backbone architecture, and Table 1 shows the detailed structure of the model.

The basic model was trained with binary cross-entropy as a loss function and the Adam optimizer [43]:

$$L_{Road}(y, p(y)) = -\frac{1}{N}\sum_{i=1}^{N}(y * \log p(y)) + (1-y) * \log(1-p(y)), \quad (1)$$

where $N$ is the total number of pixels in the image, $y$ is the true label, and $p(y)$ represents the probability of the $i$-th pixel belonging to the road and nonroad class. If $y$ is equal to 1, the $i$-th pixel belongs to the road class. When $y$ is equal to 0, the $i$-th pixel belongs to the nonroad class.

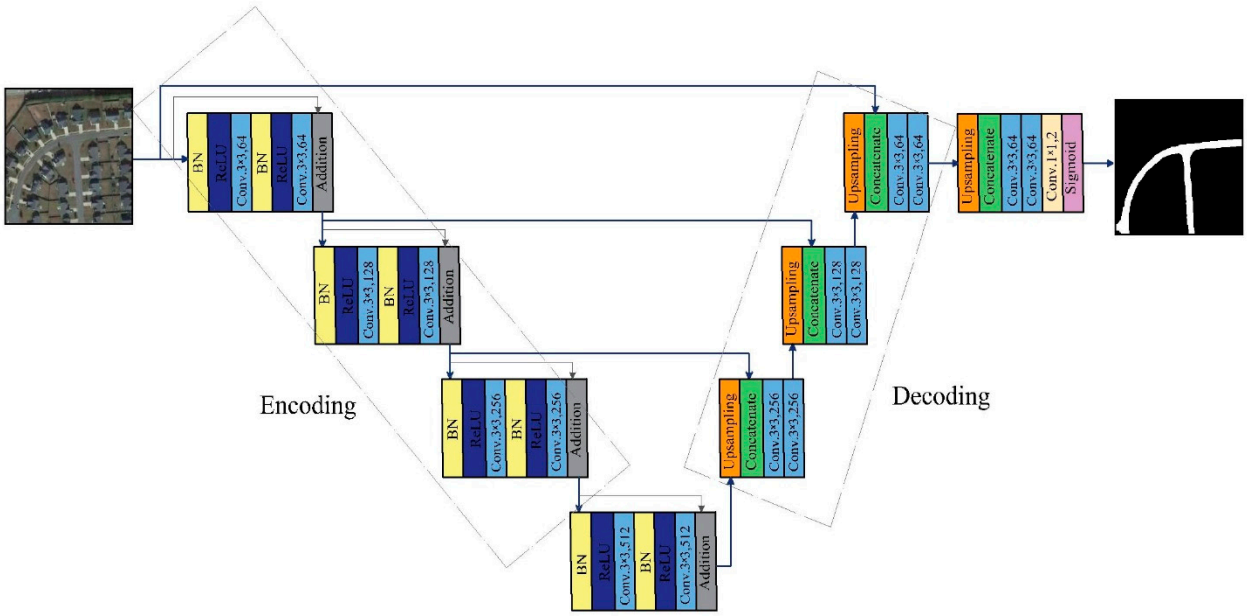


**Figure 1.** The basic architecture of the previous work by the authors [42].

**Table 1.** The network structure of our previous model [42].

| | Block Name | Filter Size | Stride | Pad | Output Size |
|---|---|---|---|---|---|
| | Input | - | - | - | 224 × 224 × 3 |
| Encoder | Res.Block1 | 3 × 3<br>3 × 3 | 2<br>1 | 1<br>1 | 112 × 112 × 64<br>112 × 112 × 64 |
| | Res.Block2 | 3 × 3<br>3 × 3 | 2<br>1 | 1<br>1 | 56 × 56 × 128<br>56 × 56 × 128 |
| | Res.Block3 | 3 × 3<br>3 × 3 | 2<br>1 | 1<br>1 | 28 × 28 × 256<br>28 × 28 × 256 |
| Bridge | Res.Block4 | 3 × 3<br>3 × 3 | 2<br>1 | 1<br>1 | 14 × 14 × 512<br>14 × 14 × 512 |
| | Conv.Block 1 | 3 × 3<br>3 × 3 | 1<br>1 | 1<br>1 | 28 × 28 × 256<br>28 × 28 × 256 |
| | Conv.Block 2 | 3 × 3<br>3 × 3 | 1<br>1 | 1<br>1 | 56 × 56 × 128<br>56 × 56 × 128 |
| Decoder | Conv.Block 3 | 3 × 3<br>3 × 3 | 1<br>1 | 1<br>1 | 112 × 112 × 64<br>112 × 112 × 64 |
| | Conv.layer1<br>Conv.layer2 | 3 × 3<br>3 × 3 | 1<br>1 | 1<br>1 | 224 × 224 × 64<br>224 × 224 × 64 |
| | Output | 1 × 1 | 1 | 1 | 224 × 224 × 2 |
| | Sigmoid | - | - | - | 224 × 224 × 2 |

### *3.2. Multitask Learning Models (MTL)*

Multitask learning refers to the approach of training a neural network to accomplish multiple tasks simultaneously. Training a neural network for MTL is a difficult process. MTL is a complicated learning process compared with STL because the role of an MTL algorithm is to harmonize and tune the learning process for multiple tasks. The elements that have huge impacts on the MTL process are an appropriate architecture for the network to ensure sharing of information among multiple tasks and a weighted loss function for the neural network. In MTL, a loss function is assigned for each task, and each loss function is summed up for multiple tasks with weights that have huge impacts on the learning algorithm. The weighted loss functions are discussed in some researches. Cipolla et al. [44] proposed an approach to dynamically weigh the loss function in MTL with respect to the homoscedastic uncertainty of the tasks. Liu et al. [45] studied the effects of negative transfer when the performance of MTL is worse than that of STL. They proposed an approach of loss balance where the weights will be updated dynamically with respect to the training process.

Based on the previous studies mentioned above, the authors have considered these factors in the architecture design and training process of the proposed models. Based on the architecture described in the previous section, this study proposes four models that have MTL training processes. Each model has a slightly different architecture of the basic backbone for STL described in the previous section.

In the next section, the detailed description of the four proposed model architectures is presented one by one—highlighting both similarities and differences in the design points within the four proposed model.

#### 3.2.1. MTL Model by Two Branches

This model represents the basic shape of the MTL model. The MTL refers to training the two tasks simultaneously, which corresponds to extracting roads and centerlines. The two tasks share the same encoder and decoder, and only the last two layers are divided into the two branches. Each branch is assigned to produce a probability map for each class. Each branch is treated as a binary task, as mentioned in the previous section. The model is

trained by binary cross-entropy as mentioned in Equation (1) for the road class. For the centerline class, Equation (2) is applied:

$$L_{Centerline}(c, p(c)) = -\frac{1}{N}\sum_{i=1}^{N}(c * \log p(c)) + (1-c) * \log(1-p(c)), \quad (2)$$

where $c$ is the true label for the centerline, and $p(c)$ represents the probability of the $i$-th pixel belonging to the centerline class. If $c$ is equal to 1, the $i$-th pixel represents the centerline. When $c$ is equal to 0, the $i$-th pixel represents a noncenterline.

The total loss function for the MTL is the summation of $L_{Road}$ multiplied by the weight ($w_{Road}$) and $L_{Centerline}$ multiplied by the weight ($w_{Centerline}$). In this study, $w_{Road}$ and $w_{Centerline}$ are set to be 1.0. The total loss function for the MTL is as follows:

$$L_{total} = L_{Road} * w_{Road} + L_{Centerline} * w_{Centerline}, \quad (3)$$

The MTL by the two branches is shown in Figure 2A. The model employs the lookahead optimizer [46] with Adam.

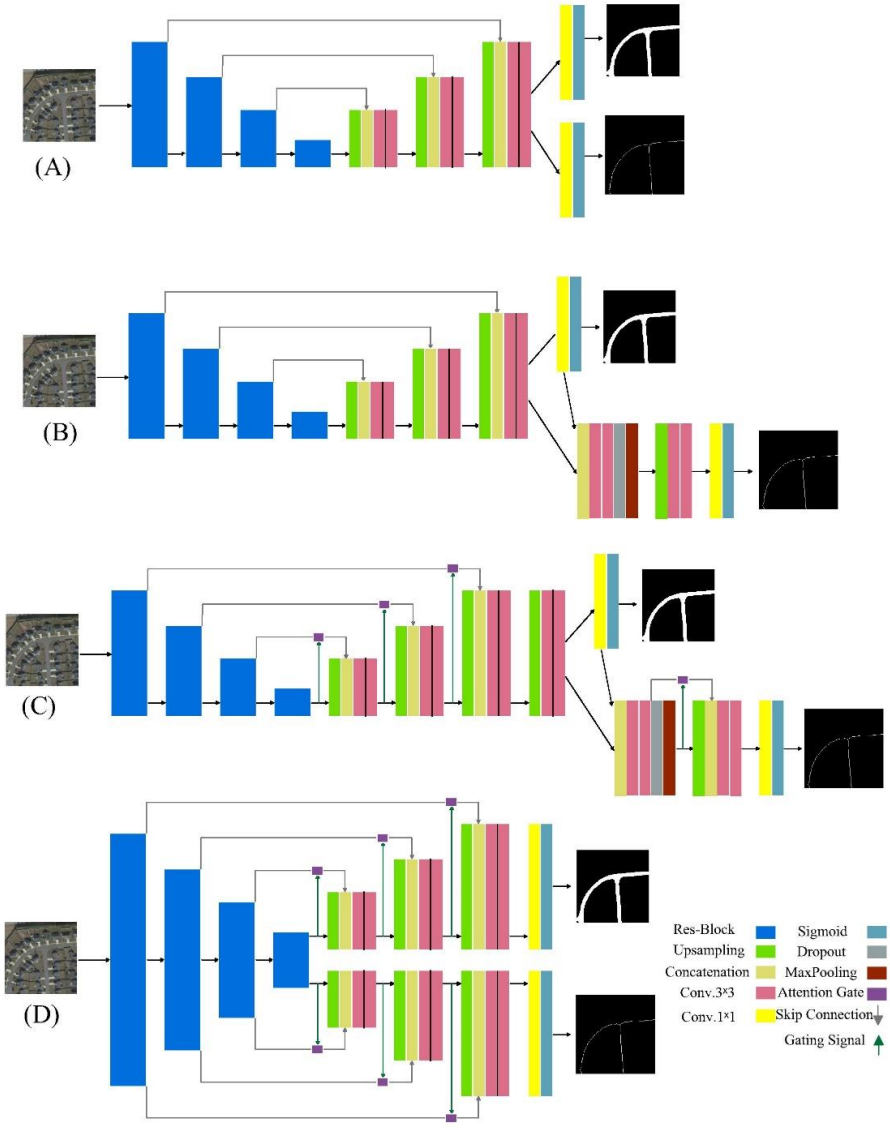


**Figure 2.** The proposed MTL models considered in this study, (**A**) MTL model by two branches, (**B**) MTL by two models, (**C**) MTL by two models with attention gates, and (**D**) MTL by one encoder and two decoders with attention gates.

#### 3.2.2. MTL by Two Models

The second MTL model consists of two models. The first extracts roads, and has the same architecture mentioned in Section 3.1. The probability map is produced by a 1×1 Conv. layer, which is fed into the second model to extract the centerline. The centerline model consists of a relatively small encoder and decoder to avoid overfitting due to the imbalanced dataset, especially for the centerline. In the encoder, a concatenate layer to connect with the 1×1 Conv.layer from the road model is employed. The two Conv.layers are employed with a dropout ratio of 0.5 and maxpooling to downsize the last feature map. In the decoder, an upsampling layer with a stride of 2 is employed to double the size of the output. Two Conv.layers and a 1×1 Conv.layer are applied to create the probability map for the centerline. This model is also trained by the binary cross-entropy, shown in Equations (1)–(3). The MTL by the two models is shown in Figure 2B.

#### 3.2.3. MTL by Two Models with Attention Gate (AG)

This model has the same architecture, as shown in Figure 2B, except for the attention gates (AGs) [22]. The AGs help to improve the segmentation process by ignoring irrelevant pixels and emphasizing relevant or desired pixels. The feature maps that have been processed and downsized in the encoder are filtered by the AGs through the U-Net skip connection. The selection process is enhanced by gating signals, which extracts contextual information from the decoder. This model has the same training process, shown in Equations (1)–(3). The MTL by two models with AG is shown in Figure 2C.

#### 3.2.4. MTL by One Encoder and Two Decoders with Attention Gate (AG)

To enhance the segmentation process, we propose another model. The model in this section consists of one encoder, and the two tasks are processed in the two decoders. There is a decoder for each task connected with the same encoder to extract the contextual information for each task simultaneously. In addition to the AG similar to the model in Figure 2C, each decoder has the same architecture as the decoder in the backbone model in Section 3.1. This model is also trained following Equations (1)–(3). The MTL by one encoder and two decoders with AG is shown in Figure 2D.

## 4. Implementation Process

In this section, we explain the datasets for roads and centerlines used in this study, data augmentation, and evaluation metrics to validate our model. Visual and quantitative comparisons of the results with the models proposed by other studies are performed in this section.

### *4.1. Dataset*

In this study, the dataset compiled by Cheng et al. [4] was used. This dataset consists of road and centerline datasets. There are 224 very high-resolution aerial images with their corresponding masks in each dataset. Cheng et al. [4] collected images from Google Earth and manually labeled the ground truth images. The images with a resolution of 1.2 m per pixel have different image sizes, and the smallest size is 600 × 600 pixels. The width of the road is 12–15 pixels, and that of the centerline is 1 pixel. Due to the high resolution, the images have a very complex background, such as the occlusion of the shadows of the buildings, trees, and cars. Hence, the extraction process for roads and centerlines is very complicated.

Each dataset is divided randomly into two sets—80% for the training set and 20% for the test set. The number of images is not enough to train a DCNN. To avoid overfitting, several methods were used in image preprocessing. First, the images were cropped to obtain the required image size to train our model (224 × 224 pixels). The cropping technique with a sliding window with a stride of 64 pixels [9,47] was employed. The second preprocessing step is to use standard data augmentation in the Keras framework. All the cropped images were augmented by shearing, zooming, random rotation, horizontal

and vertical flips, and random width and height shifts. All the models were implemented on the Keras framework on Windows 10 and one NVIDIA GeForce GTX 1070 with a learning rate of 0.0001.

### 4.2. Evaluation Metrics

Because road and centerline extraction is a segmentation task, overall accuracy (OAA) in Equation (4), intersection over union (IOU) in Equation (5), and dice coefficient (DSC) in Equation (6) are used as evaluation metrics.

$$OAA = \frac{TP + TN}{TP + TN + FP + FN}, \quad (4)$$

$$IOU = \frac{TP}{TP + FP + FN}, \quad (5)$$

$$DSC = \frac{2TP}{2TP + FP + FN}, \quad (6)$$

where $TP$, $TN$, $FP$, and $FN$ are true positive, true negative, false positive, and false negative pixels, respectively. These evaluation metrics are calculated for all the proposed models and all comparative models.

### 4.3. Comparison of the Results

In this section, we compare the four proposed models in Section 3.2 with the MTL models by other studies: cascaded net [4], RCNN-Unet2, and RCNN-Unet3 [39].

#### 4.3.1. Road Extraction

First, the results of road extraction are compared. The results are shown in Figure 3 for the visual comparison. Table 2 shows the evaluation metrics.

**Table 2.** Evaluation metrics of the models for road extraction. The results are the average of the performance for all epochs of all images in the test set. The bolded values are the highest. OAA, overall accuracy; IOU, an intersection over union; DSC, dice coefficient.

| Models | OAA | IOU | DSC |
|---|---|---|---|
| **Cascaded net** [4] | 0.9856 | 0.8053 | 0.9152 |
| **RCNN2** [39] | 0.9862 | 0.8161 | 0.9227 |
| **RCNN3** [39] | 0.9864 | 0.8171 | 0.9233 |
| **Proposed model in Figure 2A** | 0.9877 | 0.8382 | 0.9381 |
| **Proposed model in Figure 2B** | 0.9877 | 0.8381 | 0.9380 |
| **Proposed model in Figure 2C** | 0.9877 | 0.8386 | 0.9384 |
| **Proposed model in Figure 2D** | **0.9885** | **0.8492** | **0.9465** |

In images 1-(b–h), shown in Figure 3, false negative pixels (red color) are found near the intersections. The false positive pixels (blue color) in images 1-(b–f) of Figure 3 are found in the upper part of the images. In images 1-(g) and 1-(h) of Figure 3, the two models predict them successfully as road pixels (green color). The AG helps to filter the low-resolution images with a very extensive process to choose the road pixels. False positive pixels (blue color) are also found in the lower part of images 1-(d), 1-(f), and 1-(g) of Figure 3, while other models are correctly detected in this part.

In images 2-(b–h), shown in Figure 3, all the models have almost the same misseg-mented regions shown as FN pixels (red color), especially in the upper part of the images. The results of (RCNN2) 2-(c), (RCNN3) 2-(d), and 2-(f) obtained by the proposed model in Figure 2B are worse predictions for this part of the image. The proposed model in Figure 2D has successfully segmented the pixels near the intersection at the upper part of the image 2-(h) compared with the other models.

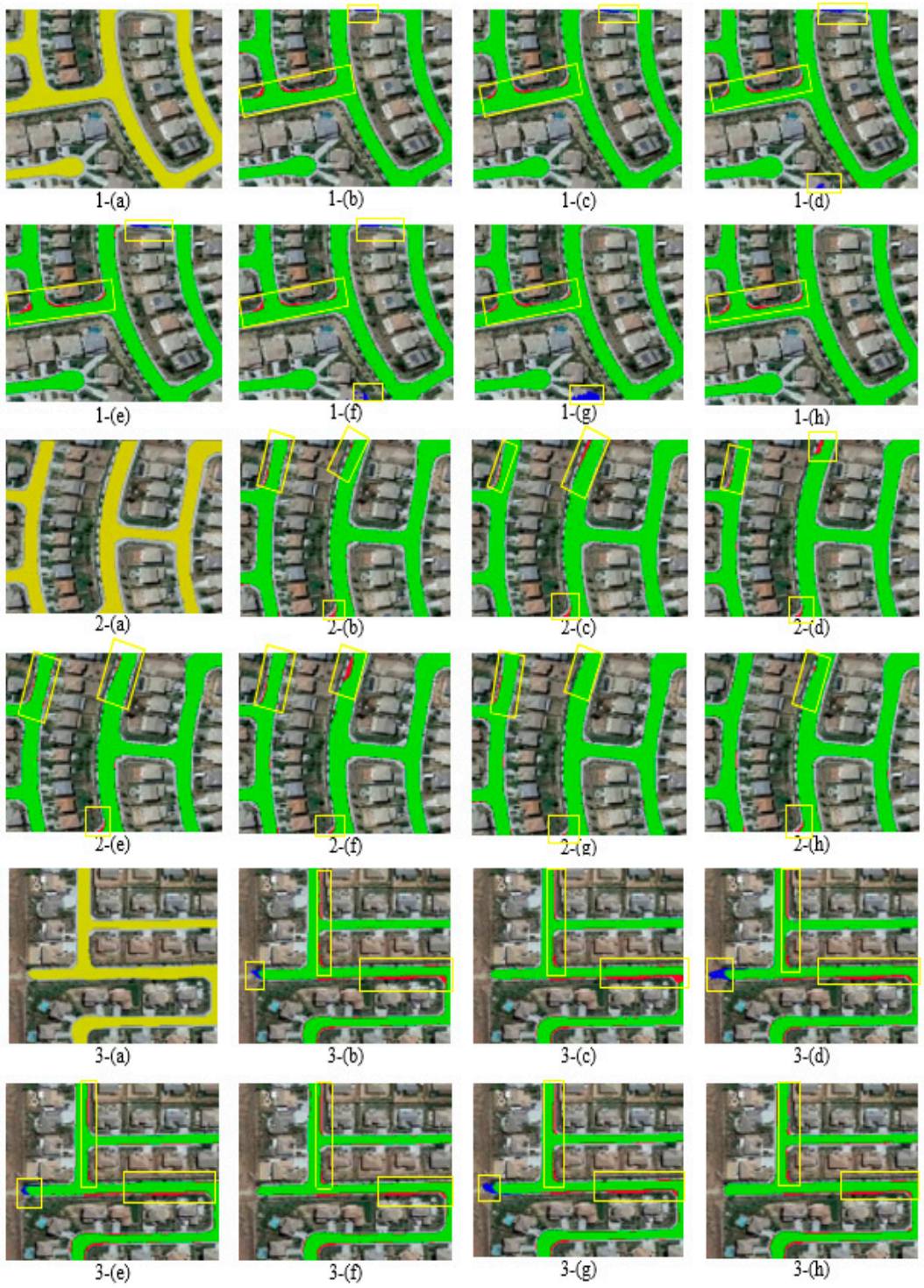


**Figure 3.** Visual comparisons for road extraction by (**a**) original image with the corresponding ground truth, (**b**) cascaded net, (**c**) RCNN2, (**d**) RCNN3, (**e**) proposed model in Figure 2A, (**f**) proposed model in Figure 2B, (**g**) proposed model in Figure 2C, and (**h**) proposed model in Figure 2D. Green, red and blue represents TP, FN, and FP, respectively.

In images 3-(b–h), shown in Figure 3, all the models have FN pixels in a similar part. For the FP pixels, we can see them clearly in images 3-(b), 3-(d), 3-(e), and 3-(g). These models miss segmented pixels as road pixels. From the visual comparison, we can find that our proposed model gives good predictions, especially for the first two images. This might be related to the good architecture combined with the attention gates, and the design of the two separate decoders has enhanced the process of segmentation compared with the other

models. According to Table 2, the best prediction was achieved by our model in Figure 2D. This model shows the highest values for all the evaluation metrics: OAA, IOU, and DSC.

#### 4.3.2. Centerline Extraction

Following the same procedure in Section 4.3.1, the results for centerline extraction are compared in Figure 4. The evaluation metrics are summarized in Table 3.

**Table 3.** Evaluation metrics of the models for centerline extraction. The results are the average of the performance for all epochs of all images in the test set. The bolded values are the highest.

| Models | OAA | IOU | DSC |
|---|---|---|---|
| **Cascaded net** [4] | 0.9754 | 0.9387 | 0.9548 |
| **RCNN2** [39] | 0.9757 | 0.9444 | 0.9578 |
| **RCNN3** [39] | 0.9759 | 0.9443 | 0.9576 |
| **Proposed model in Figure 2A** | 0.9762 | 0.9642 | 0.9726 |
| **Proposed model in Figure 2B** | 0.9763 | 0.9644 | 0.9726 |
| **Proposed model in Figure 2C** | 0.9763 | 0.9632 | 0.9716 |
| **Proposed model in Figure 2D** | **0.9767** | **0.9723** | **0.9780** |

In images 1-(b–h), shown in Figure 4, the result of the cascaded net 1-(b) has more FN pixels than those of the other models. The errors are particually found at the intersection and in the lower part of the image. Fewer FN pixels at the upper intersection are found in the results of (RCNN2) 1-(c) and (RCNN3) 1-(d), and the two proposed models 1-(f) and 1-(g). The fewest FN pixels are found in the result of image 1-(h) of the proposed model in Figure 2D. In the lower part of the image, an obvious number of FN pixels is found in all results, but the proposed model (image 1-(h)) in Figure 2D shows the best result among the other models. False positive pixels are found in the results of all the models, especially at the lower intersection. In the results of the cascaded net, false positive pixels are also found at the upper intersection.

In images 2-(b–h), shown in Figure 4, false negative pixels are found at the upper intersection, and fewer false negative pixels are found at the lower intersection for all models. The result of the cascaded net has more FN pixels in different parts of the image compared with the other models. The FP pixels are found at the upper intersection of the images for all models, and the fewest FN pixels are found in our model 2-(e).

In images 3-(b–h), shown in Figure 4, the result of the cascaded net has more FN pixels in the different parts of the image. The FP pixels are found in all results, and the fewest FP pixels are found in our model 3-(e). According to Table 3, higher values of OAA, IOU, and DSC are associated with the proposed models. The models with the two decoders with the AG give more accurate results for road and centerline extractions. Table 4 compares the evaluation metrics of our STL model [42] and our best MTL model in Figure 2D for road extraction. We can see the importance of the MTL, and the MTL model can improve the results of the STL model.

**Table 4.** Evaluation metrics of the models for road extraction performed by the authors. STL, single task learning.

| Model | OAA | IOU | DSC |
|---|---|---|---|
| **STL model** [42] | 0.9870 | 0.8465 | 0.9097 |
| **Proposed model in Figure 2D** | **0.9885** | **0.8492** | **0.9465** |

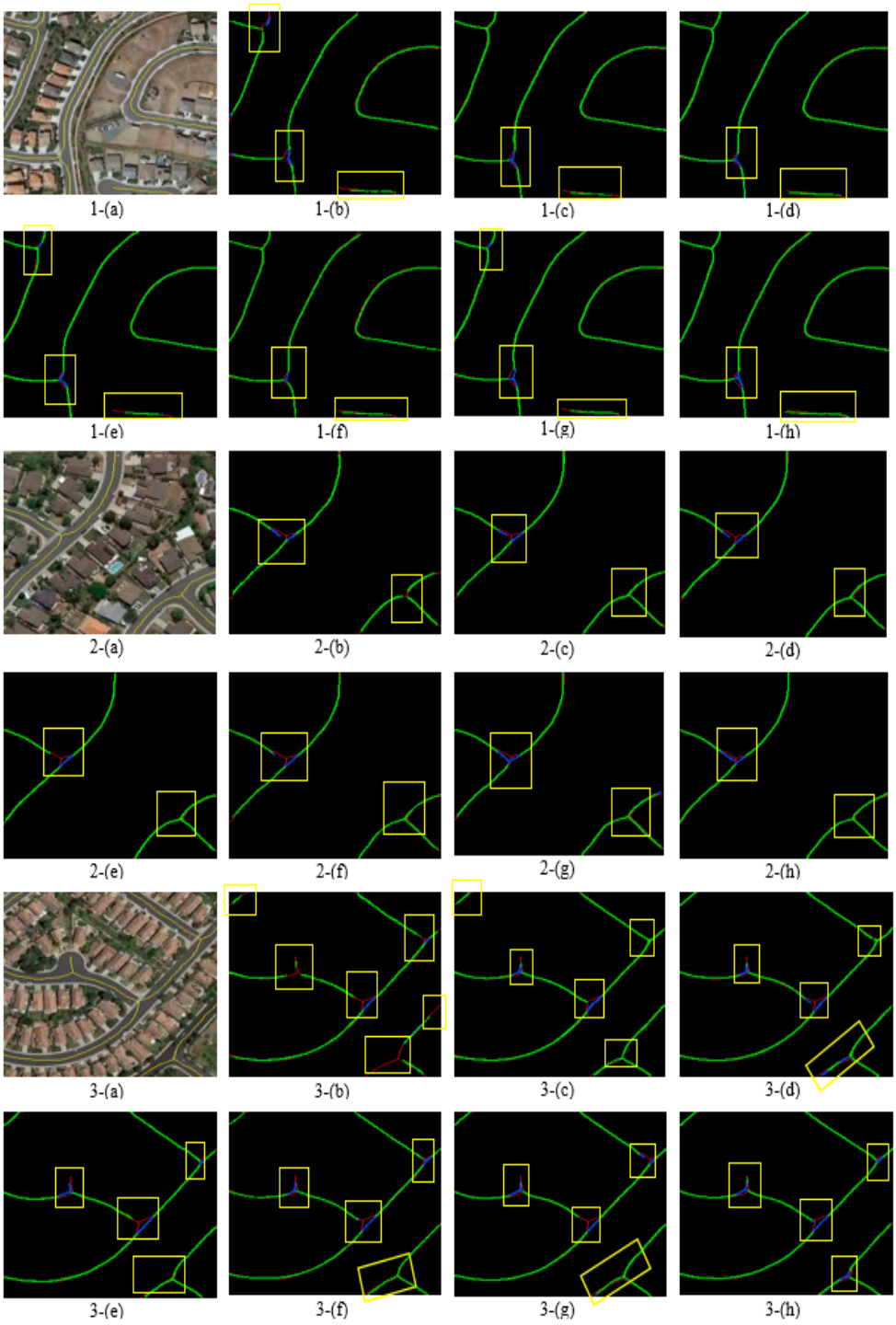


**Figure 4.** Visual comparison for centerline extraction by (**a**) original image with the corresponding ground truth, (**b**) cascaded net, (**c**) RCNN2, (**d**) RCNN3, (**e**) proposed model in Figure 2A, (**f**) proposed model in Figure 2B, (**g**) proposed model in Figure 2C, and (**h**) proposed model in Figure 2D. Green, red, and blue represent TP, FN, and FP, respectively.

## 5. Results and Discussion

The scheme of the proposed architecture tremendously influences neural networks. The proposed model, with a U-Net backbone and Res-Blocks in the encoder and Conv.layer in the decoder, was extended based on the model presented in our previous work [42], and showed the importance of a U-Net backbone and its effects on the predicted image. In MTL, the authors investigated the influence of the design of branches through the

comparison between models (A) and (B) in Figure 2. For centerline extraction, the OAA and IOU of model (B) are larger than those of model (A). Conversely, the OAA and IOU for road extraction of model (B) are a little smaller than those of model (A). This difference is because model (B) has a more detailed and deeper design for the centerline branch than model (A).

The influences of AGs are investigated by comparing the results of models (B) and (C). Model (C) was developed by adding AGs to model (B) at the main backbone of the model. The IOU and DSC for road extraction of model (C) are larger than those of model (B). However, the IOU and DSC for centerline extraction of model (C) are smaller than those of model (B). Model (C) has three additional AGs at the decoder in the branch for road extraction. On the contrary, it has only one AG at the decoder in the branch for centerline extraction. Hence, we proposed model (D) with three additional AGs for both branches symmetrically and identically. Model (D) achieved the best results. Lastly, the comparison between the STL and MTL, as shown in Table 4, shows the importance of the MTL scheme to train correlated subtasks to enhance the final segmentation results.

## 6. Conclusions and Future Work

This study presented multitask learning (MTL) models to extract roads and centerlines simultaneously, and our model successfully achieved the task using a small number of images. Our previous single task learning (STL) model for extracting only roads from aerial images was extended in this study. We proposed four models, which were developed based on the same backbone architecture of our STL model. According to the comparison of the results, using attention gates helped to enhance the selection process for the true pixels on the low-resolution feature maps developed by the encoding part of the U-Net. The two decoders were assigned to extract either road or centerline, and this network structure improved the predictions compared with the other previous models.

Although the proposed model shows good results for both tasks, the extracted centerlines are sometimes disconnected at the intersections. This issue will be studied in our future work. We will apply transfer learning, and the trained MTL model will be trained again using a new dataset. In addition, the proposed model will be trained using images with a comparatively complicated background. We will try to predict the road and centerline in the presence of shadows cast by buildings and trees. This will be addressed by employing postprocessing techniques to enhance the prediction of discontinued roads and centerlines.

**Author Contributions:** Tamara Alshaikhli: conceived the work, processed the data, and wrote the paper. Wen Liu and Yoshihisa Maruyama supervised the data processing and revised the manuscript. All authors have read and agreed to the published version of the manuscript.

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

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

## References

1. Wellmann, T.; Lausch, A.; Andersson, E.; Knapp, S.; Cortinovis, C.; Jache, J.; Scheuer, S.; Kremer, P.; Mascarenhas, A.; Kraemer, R.; et al. Remote sensing in urban planning: Contributions towards ecologically sound policies? *Landsc. Urban Plan.* **2020**, *204*, 103921. [CrossRef]
2. Wang, W.; Yang, N.; Zhang, Y.; Wang, F.; Cao, T.; Eklund, P. A review of road extraction from remote sensing images. *J. Traffic Transp. Eng.* **2016**, *3*, 271–282. [CrossRef]
3. Yao, H.; Qin, R.; Chen, X. Unmanned Aerial Vehicle for Remote Sensing Applications—A Review. *Remote Sens.* **2019**, *11*, 1443. [CrossRef]
4. Cheng, G.; Wang, Y.; Xu, S.; Wang, H.; Xiang, S.; Pan, C. Automatic Road Detection and Centerline Extraction via Cascaded End-to-End Convolutional Neural Network. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3322–3337. [CrossRef]
5. Heipke, C.; Steger, C.T.; Multhammer, R. Hierarchical approach to automatic road extraction from aerial imagery. *SPIE's 1995 Symp. OE Aerosp. Sens. Dual Use Photonics* **1995**, *2486*, 222–231. [CrossRef]

6. Bicego, M.; Dalfini, S.; Vernazza, G.; Murino, V. Automatic road extraction from aerial images by probabilistic contour tracking. In Proceedings of the 2003 International Conference on Image Processing (Cat. No.03CH37429), Barcelona, Spain, 14–17 September 2003; p. III-585. [CrossRef]
7. Zhang, Z.; Liu, Q.; Wang, Y. Road Extraction by Deep Residual U-Net. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 749–753. [CrossRef]
8. Buslaev, A.; Seferbekov, S.; Iglovikov, V.; Shvets, A. Fully Convolutional Network for Automatic Road Extraction from Satellite Imagery. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018; pp. 197–1973.
9. 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]
10. Hu, X.; Tao, C.; Hu, Y. Automatic road extraction from dense urban area by integrated processing of high imagery and LIDAR data, processing of high resolution imagery and LIDAR data. In Proceedings of the International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences IAPRSIS, Istanbul, Turkey, 23 July 2004; Volume 35, pp. 288–292.
11. Miao, Z.; Wang, B.; Shi, W.; Wu, H. A Method for Accurate Road Centerline Extraction from a Classified Image. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 4762–4771. [CrossRef]
12. Shi, W.; Miao, Z.; Debayle, J. An Integrated Method for Urban Main-Road Centerline Extraction from Optical Remotely Sensed Imagery. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 3359–3372. [CrossRef]
13. Huang, X.; Zhang, L. Road centreline extraction from high-resolution imagery based on multiscale structural features and support vector machines. *Int. J. Remote. Sens.* **2009**, *30*, 1977–1987. [CrossRef]
14. Krizhevsky, A.; Hinton, G. Image net classification with deep convolutional neural networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; Volume 1, pp. 1097–1105.
15. 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; pp. 3431–3440.
16. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
17. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P. Going deeper with convolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9.
18. Ronneberger, O.; Fischer, P.; Brox, T. U-Net. Convolutional networks for biomedical image segmentation, InMedical Image Computing and Computer-Assisted Intervention—MICCAI 2015. In Proceedings of the 18th International Conference, Munich, Germany, 5–9 October 2015; Springer International Publishing: Cham, Swizerland, 2015; Volume 9351, pp. 234–241.
19. Caruana, R.A. Multitask Learning: A Knowledge-Based Source of Inductive Bias. In *10th International Conference on Machine Learning*; Morgan Kaufmann: Burlington, MA, USA, 1993; pp. 41–48.
20. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 770–778.
21. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity Mappings in Deep Residual Networks. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Part IV. pp. 630–645.
22. Oktay, O.; Schlemper, J.; LeFolgoc, L.; Lee, M.; Heinrich, M.; Misawa, K.; Mori, K.; McDonagh, S.; Hammerla, N.; Kainz, B.; et al. Attention U-Net: Learning Where to Look for the Pancreas. *arXiv* **2018**, arXiv:1804.03999.
23. Baumgartner, A.; Carsten, S.; Helmut, M.; Wolfgang, E.; Heinrich, E. Automatic road extraction based on multi-scale, grouping, and context. *Photogramm. Eng. Remote Sens.* **1999**, *65*, 777–786.
24. Trinder, J.C.; Wang, Y. Automatic Road Extraction from Aerial Images. *Digit. Signal Process.* **1998**, *8*, 215–224. [CrossRef]
25. Poz, A.P.D.; Zanin, R.B.; Vale, G.M.D. Automated extraction of road network from medium-and high-resolution images. *Pattern Recognit. Image Anal.* **2006**, *16*, 239–248. [CrossRef]
26. Song, M.; Civco, D. Photogrammetric Engineering and Remote Sensing. *Am. Soc. Photogramm. Remote Sens.* **2004**, *12*, 1365–1371. [CrossRef]
27. Mnih, V.; Hinton, G.E. Learning to Detect Roads in High-Resolution Aerial Images. In Proceedings of the 11th European Conference on Computer Vision ECCV'10, Crete, Greece, 5–11 September 2010; Part VI. pp. 210–223.
28. Saito, S.; Yamashita, T.; Aoki, Y. Multiple Object Extraction from Aerial Imagery with Convolutional Neural Networks. *J. Imaging Sci. Technol.* **2016**, *60*, 104021–104029. [CrossRef]
29. Bastani, F.; He, S.; Abbar, S.; Alizadeh, M.; Balakrishnan, H.; Chawla, S.; Madden, S.; De Witt, D. RoadTracer: Automatic Extraction of Road Networks from Aerial Images. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Patrtern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4720–4728.
30. Mattyus, G.; Luo, W.; Urtasun, R. DeepRoadMapper: Extracting Road Topology from Aerial Images. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 3458–3466.
31. Zhang, X.; Ma, W.; Li, C.; Wu, J.; Tang, X.; Jiao, L. Fully Convolutional Network-Based Ensemble Method for Road Extraction from Aerial Images. *IEEE Geosci. Remote. Sens. Lett.* **2020**, *17*, 1777–1781. [CrossRef]
32. Guo, B.; Li, Q.; Shao, Y. Research on road centerline extraction from aerial image based on Sorting. In Proceedings of the 17th International Conference on Geoinformatics, Fairfax, VA, USA, 12–14 August 2009; pp. 1–6.

33. Sujatha, C.; Selvathi, D. Connected component-based technique for automatic extraction of road centerline in high resolution satellite images. *J. Image Video Proc.* **2015**, *8*. [CrossRef]
34. Shen, W.; Zhao, K.; Jiang, Y.; Wang, Y.; Bai, X.; Yuille, A. DeepSkeleton: Learning Multi-Task Scale-Associated Deep Side Outputs for Object Skeleton Extraction in Natural Images. *IEEE Trans. Image Process.* **2017**, *26*, 5298–5311. [CrossRef]
35. Xiao, Y.; Cai, Z.; Yuan, X. An Improved Skeleton Extraction Method via Multi-Task and Variable Coefficient Loss Function in Natural Images. *IEEE Access* **2019**, *7*, 171272–171284. [CrossRef]
36. Liu, P.; Qiu, X.; Huang, X. Adversarial Multi-task Learning for Text Classification. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July–4 August 2017. [CrossRef]
37. Chen, C.; Bai, W.; Rueckert, D. Multi-task Learning for Left Atrial Segmentation on GE-MRI. In *Statistical Atlases and Computational Models of the Heart. Atrial Segmentation and LV Quantification Challenges. STACOM 2018. Lecture Notes in Computer Science*; Pop, M., Ed.; Springer: Cham, Switzerland, 2019; Volume 11395, pp. 292–301. [CrossRef]
38. Gonçalves, G.R.; Diniz, M.A.; Laroca, R.; Menotti, D.; Schwartz, W.R. Multi-task Learning for Low-Resolution License Plate Recognition. In *Proceedings in Pattern Recognition, Image Analysis, Computer Vision, and Applications. CIARP 2019. Lecture Notes in Computer Science*; Nyström, I., Hernández Heredia, Y., Milián Núñez, V., Eds.; Springer: Cham, Switzerland, 2019; Volume 11896, pp. 251–261.
39. Yang, X.; Li, X.; Ye, Y.; Lau, R.Y.K.; Zhang, X.; Huang, X. Road Detection and Centerline Extraction via Deep Recurrent Convolutional Neural Network U-Net. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7209–7220. [CrossRef]
40. Liu, Y.; Yao, J.; Lu, X.; Xia, M.; Wang, X.; Liu, Y. RoadNet: Learning to Comprehensively Analyze Road Networks in Complex Urban Scenes from High-Resolution Remotely Sensed Images. *IEEE Trans. Geosci. Remote Sens.* **2018**, *57*, 2043–2056. [CrossRef]
41. Shao, Z.; Zhou, Z.; Huang, X.; Zhang, Y. MRENet: Simultaneous Extraction of Road Surface and Road Centerline in Complex Urban Scenes from Very High-Resolution Images. *Remote Sens.* **2021**, *13*, 239. [CrossRef]
42. Alshaikhli, T.; Liu, W.; Maruyama, Y. Automated Method of Road Extraction from Aerial Images Using a Deep Convolu-tional Neural Network. *Appl. Sci.* **2019**, *9*, 4825. [CrossRef]
43. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference for Learning Representations, San Diego, CA, USA, 5–8 May 2015.
44. Kendall, A.; Gal, Y.; Cipolla, R. Multi-task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 19–21 June 2018; pp. 7482–7491. [CrossRef]
45. Liu, S.; Liang, Y.; Gitter, A. Loss-Balanced Task Weighting to Reduce Negative Transfer in Multi-Task Learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 9977–9978.
46. Zhang, M.; Lucas, J.; Hinton, G.; Ba, J. Lookahead Optimizer: K steps forward, 1 step back. *arXiv* **2019**, arXiv:1907.08610.
47. Wang, H.; Wang, Y.; Zhang, Q.; Xiang, S.; Pan, C. Gated Convolutional Neural Network for Semantic Segmentation in High-Resolution Images. *Remote Sens.* **2017**, *9*, 446. [CrossRef]