Study on the effect of color space in deep multitask learning neural networks for road segmentation

Jere Raninen, Lingli Zhu, Emilia Hattula

National Land Survey of Finland (NLS) -jere.raninen@nls.fi, lingli.zhu@nls.fi, emilia.hattula@nls.fi

Keywords: Road Segmentation, Aerial Imagery, Color Space, Deep Learning, Neural Network.

Abstract

Precise road segmentation is an essential part of many applications related to road information extraction from remote sensing data. The effect of color space on road detection has rarely been studied. In this paper, the effects of different color spaces of aerial images and multitask learning methods were experimented on road segmentation using three deep convolutional neural networks, UNet, DenseU-Net, and RoadVecNet. The color spaces included RGB, HSV, LAB, YCbCr, and YUV. The multitask learning methods adopted in this study involved utilizing multiple inputs, and multiple outputs. Multiple inputs were aerial images from the same area with different color spaces, and multiple outputs were road segmentation and road outline segmentation. As remote sensing data, National Land Survey of Finland's true orthophotos (from 2020), Massachusetts road imagery dataset, and Ottawa dataset were applied. Segmentation masks for National Land Survey of Finland's true orthophotos were extracted from Digiroad vectors with road width information. Road outline masks were generated from the segmentation masks. The studied neural networks were trained with the same data, learning rate, loss function, and optimizer for each color space, and pairs of color spaces. Multiple outputs were experimented with RGB color space. The comparative analysis assessed the performance of various neural networks across different color spaces using the F1-score metric. The experimental findings indicate that the choice of color space has little influence on the results of neural networks Deep learning methods can adapt to different color spaces well. In addition, the use of sharpening and edge enhancement augmentations had a slight effect on the results.

1. Introduction

Accurate road segmentation has applications, such as road navigation, urban infrastructure development, and geographic information collection. Although there have been many studies (Bastani, 2018; Liu, 2018; Mnih, 2013) on road segmentation, the challenge remains due to noise in the datasets, complex image background, and occlusions from structures, such as buildings, trees, vehicles, and shadows.

Deep convolutional neural networks have emerged as an effective approach for image analysis and detail extraction tasks, including road segmentation (Abdollahi, 2021). Notably, encoder-decoder networks based on UNet-architecture have proven useful in addressing these challenges (Abdollahi, 2021; Dong, 2019; Henry, 2021; Ronneberger, 2015).

This study extends the findings presented in (Raninen, 2022), where different color spaces were explored for neural network-based road segmentation. In this study, we incorporate additional datasets and data augmentation methods to assess the efficiency of different color spaces in road segmentation tasks using deep learning neural networks. The study explores various color spaces across different neural network architectures, conducting tests on three distinct datasets. The performance of UNet (Ronneberger, 2015), DenseU-Net (Dong, 2019), and RoadVecNet (Abdollahi, 2021) are experimented with different color spaces. The color spaces include RGB, HSV, LAB, YCbCr, and YUV. Additionally, the effect of edge enhancing and sharpening operations are tested on each color space to see the effect of those operations with different color spaces. The experiments include training each model with aerial images of each color space and each pair of these color spaces. The models are trained for road segmentation tasks. Additionally, RGB color space is experimented with additional output, making the model produce both road surface segmentation and road outline segmentation simultaneously. Additional inputs are tested by adding an encoder and skip-connections to the network, and additional outputs are tested by adding a smaller encoder-decoder pair at the end of the original model, feeding it the concatenation of original input and the output of the first encoder-decoder pair. The results are compared with F1-score, and conclusions are drawn.

This paper aims to answer what kind of neural networks can be used for efficient and accurate road segmentations, can the choice of color space of the input images affect the performance of the neural network? Can edge enhancement and sharpening operations improve the results on each color space? And can multitask learning methods improve the performance of neural networks in road segmentation?

2. Related work

The basic idea of CNN was introduced by (Fukushima & Miyake, 1982) as a neural network model for visual recognition tasks. (LeCun, 1989) were able to successfully use CNN in hand-written digit recognition tasks.

Deep Convolutional Neural Networks were pioneered by (Krizhevsky, 2012). Deep CNNs were originally used for image classification tasks, being effective in extracting deep features from images, but Deep CNNs still had problems in semantic segmentation tasks. (Long, 2015) introduced fully convolutional neural networks (FCN) which achieved better results in image segmentation tasks. (Ronneberger, 2015) developed the UNet model for medical image segmentation. (Bastani, 2018) proposed RoadTracer to construct roads from aerial images. They used a CNN-based decision function to process the output of CNN. (Dong, 2019) proposed the DenseU-Net model. It is an

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-4-W4-2024-201-2024 | © Author(s) 2024. CC BY 4.0 License.

end-to-end FCN that is based on the UNet. DenseU-Net uses cascade operations to combine the CNN features (Dong, 2019). (Abdollahi 2021) introduced RoadVecNet which uses two interlinked UNet-based networks to do both road segmentation and road vectorization tasks with a single model. The first UNet does the segmentation task and the second UNet uses the output of the first UNet to produce the road vectors. (Xu, 2022) proposed transformer-based network RNGDet to generate road network graph using aerial images. (Ozturk, 2023) used feature-wise fusion with neural network to combine optical images and point clouds, improving the road segmentation performance of the model. In our previous study (Raninen, 2022), different color spaces were evaluated for their effectiveness in road segmentation tasks using UNet, DenseU-net, and RoadVecNet architectures. The findings indicated minor benefits when using certain color spaces. This foundational work provided basis for the current study, which aims to further explore the impact of color spaces by incorporating additional datasets and employing sharpening and edge enhancement.

(Caruana, 1997) coined the term multitask learning in his paper "Multitask learning". In the paper, Caruana explained multitask learning as an inductive transfer mechanism which aims to improve the generalization of a neural network by using the domain-specific information contained in the training signals of related tasks as an inductive bias by learning tasks in parallel while using a shared representation. In other words, multitask learning uses additional, related tasks to help the model learn the main task better (Caruana, 1997). (Saito, 2016) Introduced a method to produce multiple predictions and use multiple labels to train CNNs. (Jha, 2020) integrated semantic image segmentation and depth estimation as different tasks during training, thereby enhancing the overall performance of the network by jointly capturing both scene semantics and geometry.

(Jurio, 2010) compared different color spaces in clustering-based image segmentation.

3. Data

Three datasets are used in this paper for road segmentation, one produced by the National Land Survey of Finland (NLS), Massachusetts road imagery dataset (Mnih, 2013), and Ottawa dataset (Liu, 2018). NLS dataset includes orthophotos and road labels, Massachusetts dataset include aerial images and road labels, while Ottawa dataset includes google earth images and road labels. Different color spaces were generated from the RGB-images of the datasets and labels for road outline segmentation were generated from the road surface labels.

3.1 NLS dataset

The NLS dataset includes orthophotos from different regions of Finland and corresponding road centerline vectors. The Orthophotos were produced from aerial images and Digital Elevation Model (DEM). Orthophotos removed the image distortions, including radiometric and geometric distortions. NLS Dataset used in this paper were produced by National Land Survey of Finland. The Data has been collected by NLS by flying over the whole area of Finland over multiple years while taking aerial images and aerial lidar (National Land Survey of Finland, 2021). The aerial images are accessible on the website of National Land Survey of Finland.

NLS Orthophotos used in this study contains orthophotos covering different regions of Finland, including areas of Central Finland, Kainuu, North Ostrobothnia, Southern Ostrobothnia, Ostrobothnia, Pirkanmaa, Päijät-Häme, Southwest Finland, and Uusimaa. 24 areas were chosen from these regions covering about 36km^2 each. In our case, the aerial images were acquired from the year 2020. The spatial resolution, which is the physical dimension that represents a pixel of the image, of NLS orthophotos is 50cm (National Land Survey of Finland, 2021). Orthophoto contains x, y, R, G, and B information. The orthophotos have a size of 12000 x 12000px, that is, 6000m x 6000m. The orthophotos are in the form of GeoTiff, with the coordinate system of ETRS89-TM35FIN (National Land Survey of Finland, 2021).

In the experiments the orthophotos were cropped to 1000 x 1000 pixels without any overlapping, and before using the images as input, the images were randomly cropped further to 512 x 512 pixels.

Road labels were from the Digiroad (Väylävirasto, 2021) road vectors, which included the road centerline vector with width information on most of the roads. For roads without width, smallest width from the dataset was used. The labels contain both road surface and road outline segmentation masks. Road outlines were used as an additional output in the experiments with multitask learning networks.

Both the road surface and road outline segmentation masks were generated using Python library OpenCV. The masks were generated by drawing the road vectors and buffering the vectors with the corresponding width attribute, or smallest width in the cases where the width attribute was missing. The road outlines were generated from these surface segmentation labels by dilating the road surface masks once with 3x3 dilation and subtracting the original image array from the dilated image resulting in the outlines of the roads.

The final dataset consists of orthophotos from NLS (2020), segmentation masks of roads, and road outline masks. The dataset also contains images in different color spaces, including RGB, HSV, LAB, YCbCr, and YUV. The different color spaces were generated from RGB images with OpenCV Python library. The dataset contains 1583 1000x1000px aerial images that were randomly split into train, test, and validation sets. Train set contains 75%, test set 15%, and validation set 10% of all the images. There was no overlapping between the images, and thus the training set, test set, and validation set all have unique images.

NLS dataset contains both urban and rural areas. The roads in the dataset have occlusions by buildings, vehicles, trees, and shadows as well as different surfaces and widths, overpasses, underpasses, bridges, parking lots, and roundabouts, making it a difficult dataset for road segmentation. Example images and labels of NLS dataset can be seen in *Figure 1*.

Figure 1: NLS dataset example images, road surface segmentation masks, and road edge masks

Producing high-quality pixel-level labels is costly, and even manually produced labels often contain topological errors. This dataset is no exception, containing similar inaccuracies in labels that (Henry, 2021) describes in their paper, including omission noise, registration noise, and over-simplification of the labels. These errors are dealt with by choosing a noise-aware loss as the loss function of the neural network. In our case, the combination of soft-bootstrapped binary cross-entropy loss and Dice-coefficient loss (Henry, 2021) was chosen.

Omission noise means that the annotator misses some objects of interest (roads in this case). Registration noise, on the other hand, means that the label is not exactly where it is supposed to be, missing the target object partially or offsetting it by some amount. Finally, the over-simplification of the labels means that the labels are overly simple, for example, the thickness of the road is annotated with fixed thickness per road and the centerline of the road is not exact.

Label noise affects the training of the network, making it harder to train, while making the benchmark less reliable, since good predictions can be penalized because of the faulty ground-truths. (Henry, 2021) recommends using noise-aware losses as one solution to these problems. Noise-aware losses re-balance the confidence granted to the ground-truth in favor of the predictions (Henry, 2021).

3.2 Massachusetts road imagery dataset

Massachusetts road imagery dataset (Mnih, 2013) contains aerial images with 0.5 m spatial resolution and road labels.

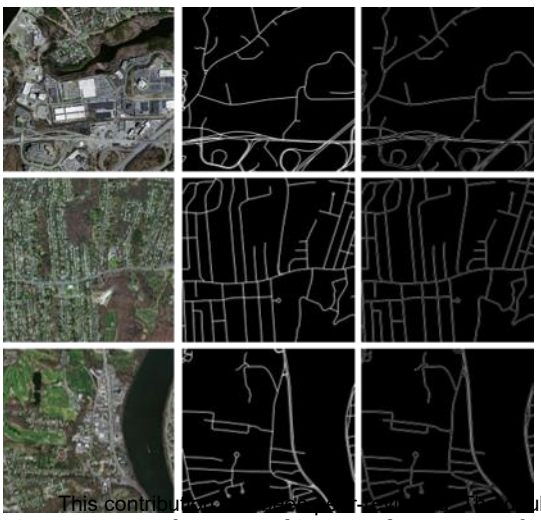

Figure 2: Massachusetts road imagery dataset example images, road surface segmentation masks, and road edge masks

Aerial images have roads with different widths and complex backgrounds. Dataset covers an area of over 2600 square kilometres, containing urban, suburban, and rural regions.

The dataset was randomly split into training, test, and validation sets. Training set consists of 1171 1500x1500px aerial images, while validation set consists of 14 images and test set 49 images of the same size. Some of the training images contained large white areas with pixels of zero-value. From these, images with over 40% zero-pixels were removed resulting in 962 training images.

Each image has a road surface segmentation label. Road outline labels were generated from these road surface segmentation labels. Example images and labels of Massachusetts road imagery dataset (Mnih, 2013) can be seen in Figure *2*.

Original images were RGB-images, and like NLS dataset, images with different color space were generated from these. Color spaces include RGB, HSV, LAB, YCbCr, and YUV.

3.3 Ottawa dataset

Ottawa Dataset (Liu, 2018) contains several urban areas of Ottawa, Canada. The images are Google Earth images with spatial resolution of 0,21m covering 21 regions of about 8km^2. The dataset is split into training, test, and validation sets based on these regions. Training set contains 14 regions, validation set contains one region, and test set contains six regions The dataset contains manually annotated road surfaces, road edges, and road centerlines for each image. There are roads of different width, from 10px to 80px, and the roads in the images contain occlusions because of, for example, shadows cars and trees (Liu, 2018). Example images and labels of Ottawa dataset (Liu, 2018) can be seen in ***Figure 3***.

As with the other datasets, different color spaces were also generated from the Ottawa RGB images, including HSV, LAB, YCbCr, and YUV.

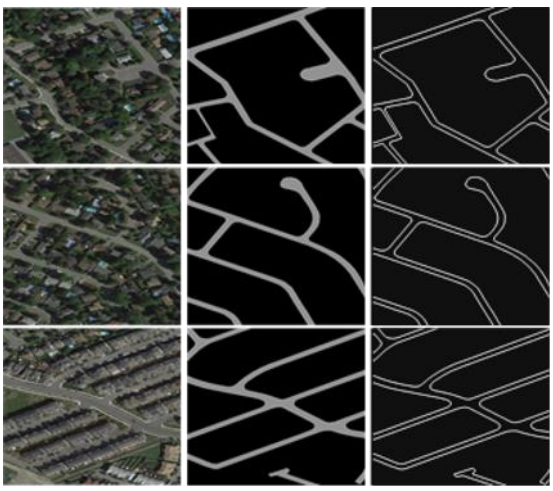

Figure 3: Ottawa dataset example images, road surface segmentation masks, and road edge masks

.

4. Methods

First, we describe the neural networks UNet, DenseU-Net, and RoadVecNet. Finally, the training process is described. The neural networks were implemented using TensorFlow Keras version 2.8.0 on Python programming language (version 3.7).

4.1 UNet

UNet was introduced by (Ronneberger, 2015) for medical image segmentation. The UNet (Ronneberger, 2015) used in these experiments was modified from the original paper by adding padding to each convolution to keep the resolution of the output same as the resolution of the input. Also, the bottleneck, which is the layer between encoder and decoder, used dropout with the rate of 0.5 to help the model in avoiding overfitting.

The UNet (Ronneberger, 2015) contains input layer, with input size of (512, 512, 3) since the input images are 512x512px and contain three color channels. The encoder consists of four convolution blocks, each having a 3x3 convolution layer, ReLU-activation (rectified linear unit), 3x3 convolution layer, ReLU-activation, and finally a MaxPooling layer with 2x2 pool size. Outputs of the last ReLU-activations from each convolution block were used in the skip-connections. The decoder part, on the other hand, consists of an upsampling of the feature map, 2x2 convolution that halves the number of feature channels, concatenation layer which combines the feature map with the corresponding feature map from the encoder(s) via skip-connections, and two 3x3 convolution layers, both followed by a ReLU-activation. The final layer is a 1x1 convolutional layer with SoftMax-activation that produces the final segmentation (Ronneberger, 2015).

4.2 DenseU-Net

DenseU-Net was introduced by (Dong, 2019). It is an end-to-end FCN that is based on the UNet (Ronneberger, 2015). DenseU-Net uses cascade operations to combine the CNN features (Dong, 2019). The structure is composed of multiple DownBlocks (down-sampling block) and UpBlocks (up-sampling block). The context information is generated with the DownBlocks, and the features are restored back to the original image resolution with UpBlocks. Like UNet (Ronneberger, 2015), DenseU-Net also consists of encoder and decoder, and skip connections are used to connect the shallow encoder layers with the deeper decoder layers (Dong, 2019).

The encoder consists of five consecutive DownBlocks. After each DownBlock, the number of feature dimensions double. Each DownBlock gets D-dimensional Height x Width feature maps as input. The input goes through two sets of convolutions, batch normalization (BN) (Ioffe & Szegedy, 2015), ReLU-activation, and concatenation. The convolutional layers both use D-dimensional kernels with a filter size of three and a stride of one. The concatenation layers combine the input and all the previous results inside the DownBlock together. Next the result goes through another convolutional layer, batch normalization (Ioffe & Szegedy, 2015) and ReLU-activation, but this time the convolutional layer has a filter size of one. This result is both shared with the corresponding UpBlock via the skip connection and used as an input for the final layer of the DownBlock, a MaxPooling layer.

The decoder, on the other hand, consists of five consecutive UpBlocks. After each UpBlock, the number of feature dimensions halves. There are also skip connections that connect the corresponding DownBlock to UpBlock with identical resolution. Each UpBlock takes similar input to DownBlocks, D-dimensional height x width feature map. The UpBlock consists of a transposed convolutional layer, concatenation layer, and four convolutional layers. First, the input feature maps are upsampled with transposed convolutional layer with a stride of two doubling the height and width of the feature maps. Then the upsampled feature maps are concatenated with the feature maps from the corresponding DownBlock that have same resolution to the upsampled feature maps via skip-connection. Then the feature maps are reduced back to D-dimensions with a D-dimensional convolutional layer with a filter size of one. Next, the feature maps go through two convolutional layers with D-dimensional kernels, filter size of three, and stride of one. The output of the first convolutional layer is combined with the input of the first convolutional layer and the output of the second convolutional layer is combined with the input and output of the first convolutional layer via concatenation layers. Finally, the output feature maps go through a final convolutional layer with a D-dimensional kernel and a filter size of one. This way the output is reduced back to D-dimensional after the concatenation. ReLU-activation and a batch normalization layer (Ioffe & Szegedy, 2015) come after each convolutional layer and transposed convolutional layer in the UpBlock. The final layer is a SoftMax layer that is used for predicting the output segmentation. In this study, DenseU-Net (Dong, 2019) and UNet (Ronneberger, 2015) were trained with initial filter size of 64.

4.3 RoadVecNet

(Abdollahi, 2021) introduced RoadVecNet that uses two interlinked UNet-based networks to do both road surface segmentation and road vectorization with one model. The first UNet does the segmentation task and the second UNet uses the input and output of the first UNet to produce the road outlines.

The segmentation model uses pre-trained VGG-19 (Visual Geometry Group) (Simonyan & Zisserman, 2014) as the encoder. Between the encoder and decoder RoadVecNet has a dense dilated spatial pyramid pooling (DDSPP) module (Yang, 2018). It helps in extracting high-resolution feature maps and capture contextual information (Abdollahi, 2021). DDSPP module (Yang, 2018) consists of several dilated convolutional layers followed by concatenation layer that combines the input and all the previous outputs produced inside the DDSPP together. Four dilated convolutional layers and four concatenation layers were used in this study. The dilation rate of each layer from first to last were 2, 4, 8, and 12 (Abdollahi, 2021).

The segmentation decoder consists of four decoder blocks. Each decoder block contains upsampling layer, concatenation layer that concatenates the output of the upsampling layer with the corresponding skip connection, then two 3x3 convolutional layers, batch normalization (Ioffe & Szegedy, 2015), ReLU-activation, and finally a squeeze-and-excite (SE) module (Hu, 2018). The upsampling layer is a 2x2 bilinear upsampling layer that is used to double the dimensions of the input feature map. The BN is used to stabilize the network. It standardizes the inputs to a layer in the network and can increase the training speed of a network (Abdollahi, 2021). The SE module is used to pass more relevant data and reduce redundant ones (Abdollahi, 2021). After the decoder comes the output-block, which consists of 1x1 convolutional layer with the number of filters equal to the number of segmentation classes, and finally a SoftMax activation layer that produces the segmentation mask.

4.4 Training

The models were trained for 60 epochs with a learning rate of 1e-4 with the Adam optimizer (Kingma & Ba, 2014). Batch size of 2 is used, and images are normalized between values 0-1. Data augmentations used in the experiments include random horizontal and vertical flipping, and random cropping to 512x512px. Additionally, edge enhancement and sharpening operations are experimented with each color space with RoadVecNet model (Abdollahi, 2021).

Experiments include 1) Each model is trained with each color

space for both datasets, 2) RoadVecNet is additionally trained twice with each color space, first with edge enhancement and second with sharpening for each dataset, 3) each model is trained with a pair of color spaces for each dataset, 4) each model is trained in each dataset in RGB color space with two outputs: road surface segmentation, and road outline segmentation.

With two inputs, the network was trained with two encoders, one for each input, and one decoder, which got the concatenated outputs of each encoder as its input and combined the skip-connections from each encoder.

With multiple outputs, smaller version of the model, with 1 block less in the encoder and in the decoder compared to the original model, was added at the end of the model to produce the road outline segmentation. This was done in a similar manner as described in (Abdollahi, 2021). The second network gets a concatenation of the original input and the output of the first network as its input, finally producing the road outlines.

The different neural networks, UNet (Ronneberger, 2015), DenseU-Net (Dong, 2019), and RoadVecNet (Abdollahi, 2021), were trained to perform road surface segmentation from aerial images. The networks were trained on five different color spaces, including RGB, HSV, LAB, YCbCr, and YUV. Color spaces were used as a single-input and as multi-input, meaning two different color spaces simultaneously. The networks were also trained to produce both a single output and multiple outputs. Road surface segmentation was used as a single output, and both road surface segmentation and road outline segmentation were used as multiple simultaneous outputs. The models with multiple outputs were trained in the RGB color space. The results between different models, color spaces, and tasks are compared to see the effect of additional outputs on the performance of the networks.

Additionally, RoadVecNet (Abdollahi, 2021) is trained with additional augmentations, edge enhancement and sharpening, in different color spaces to see if color space affects the performance of these augmentations and if these augmentations can be used to improve the model accuracy in road segmentation. Edge enhancement and sharpening are applied with Pillow Python image processing library. Sharpening sharpens the image along edges, and edge enhancement enhances the contours of the image (Taylor & Nitschke, 2018).

Because the labels are noisy, we decided to follow (Henry, 2021) and use a combination of soft-bootstrapped binary cross-entropy loss and Dice-coefficient loss to reduce the negative effects of the noise in the labels. In the experiments, F1-score was used to evaluate the performance of the models. F1-score measures the closeness of the predicted mask to the ground truth mask. Mathematically, F1-score is calculated using the formula in equation 1,

$$F1 = \frac{2 * Precision * Recall}{Recall + Precision} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

where precision and recall are introduced in equations 2 and 3. Precision means the ratio of correctly predicted true positives from the total number of positives predicted, and recall means the ratio between correctly predicted positives against the total number of actual positives. TP stands for true positives, meaning the pixels that are correctly classified as positive, FP stands for false positives, that is pixels incorrectly classified as positive, and FN stands for false negatives, which mean pixels that were incorrectly classified as negative.

5. Results

5.1 Single color space

The results of UNet, DenseU-Net, and RoadVecNet trained on each color space with NLS dataset can be seen in **Table 1**, with Massachusetts dataset in Table 2, and with Ottawa dataset in Table 3. Generally, RGB color space had the best performance. Every model performed worst with NLS and Ottawa dataset in YUV color space.

	UNet	DenseUNet	RoadVecNet
RGB	68,4	69,7	71
HSV	68,5	69,2	69,8
LAB	68,3	69,4	70,1
YCbCr	68,4	69,4	70,4
YUV	68	69,2	69,8

Table 1: F1-scores in different color spaces with NLS dataset.

	UNet	DenseUNet	RoadVecNet
RGB	77,05	79,14	80,4
HSV	62,29	77,58	80,48
LAB	49,13	78,07	77,21
YCbCr	77,18	78,38	76,82
YUV	74,5	78,09	77,41

Table 2: F1-scores in different color spaces with Massachusetts dataset.

	UNet	DenseUNet	RoadVecNet
RGB	93,9	95,4	95,9
HSV	93,5	95,5	95,2
LAB	93	95,5	95
YCbCr	82,2	95,5	95,7
YUV	82	95,4	94,7

Table 3: F1-scores in different color spaces with Ottawa dataset.

DenseU-Net performed very similarly with each color space. The difference between the best (0.955 F1-score with Ottawa dataset, and 0.697 F1-score with NLS dataset) and the worst (0.954 F1-score with Ottawa dataset, and 0.692 F1-score with NLS dataset) performance with DenseU-Net is 0.001 F1-score with Ottawa dataset and 0.005 F1-score with NLS dataset.

Even though RoadVecNet used encoder that was pretrained in RGB color space with ImageNet (Deng, 2009), it achieved almost as good performance with YCbCr color space, and slightly better F1-score in Massachusetts dataset with HSV color space.

UNet got stuck in local minima with Ottawa dataset in YCbCr and YUV color spaces and with Massachusetts dataset in HSV and LAB color spaces, achieving much worse results compared to the other results in Ottawa dataset or Massachusetts dataset. This could be due to the use of dropout layer with 0.5 dropout rate. With NLS dataset, UNet achieved very similar results with each color space. Worst performance was with YUV color space (0.68 F1-score), and best performance was with HSV color space (0.685).

All things considered; the choice of color space doesn't affect the performance of road segmentation much. RGB color space is a good choice.

5.2 Additional augmentations

The results of RoadVecNet trained in different color spaces without sharpening or edge enhancement, with sharpening, and with edge enhancement on NLS dataset can be seen in Table 4, with Massachusetts dataset in **Table 5**, and with Ottawa dataset in Table 6. Generally, the results show that sharpening and edge enhancement improve the performance of road surface segmentation with some color spaces on each dataset, and edge enhancement performs slightly better than sharpening. Edges and contours are important visual cues for road segmentation from aerial images and sharpening and enhancing them to be more visible can affect the results.

	Original	Sharpen	Edge Enhance
RGB	71	71,3	70,9
HSV	69,8	70	70,5
LAB	70,1	70,3	70,7
YCbCr	70,4	70,6	70,8
YUV	69,8	70,6	70,6

Table 4: F1-scores without additional augmentations, with sharpening, and with edge enhance in different color spaces with NLS dataset.

	Original	Sharpen	Edge Enhance
RGB	80,4	80,65	79,59
HSV	80,48	78,12	78,28
LAB	77,21	79,15	79,24
YCbCr	76,82	77,19	78,72
YUV	77,41	77,82	79,01

Table 5: F1-scores without additional augmentations, with sharpening, and with edge enhance in different color spaces with Massachusetts dataset.

	Original	Sharpen	Edge Enhance
RGB	95,9	96,3	95,5
HSV	95,2	95,2	95,5
LAB	95	95,4	96
YCbCr	95,7	95,2	95,6
YUV	94,7	94,8	95,6

Table 6: F1-scores without additional augmentations, with sharpening, and with edge enhance in different color spaces with Ottawa dataset.

The results show that edge enhancement improves the results more than sharpening in all color spaces except for two cases; RGB and YCbCr color space. In RGB color space, sharpening yields the best result. In YCbCr color space, the Ottawa dataset has the best result without sharpening or edge enhancement, but with both NLS dataset and Massachusetts dataset results with sharpening and edge enhancement have better F1-score than without additional augmentation.

In RGB color space, sharpening performs better on each dataset. YCbCr has differing results with Ottawa dataset, where sharpening and edge enhancement performed slightly worse compared to the performance without additional augmentations. Since the models were trained only once, these results could be due to randomness. The F1-scores are still close to the original values and the effect of these augmentations is small.

5.3 Multiple inputs

The results of UNet, DenseU-Net, and RoadVecNet trained on each color space pair with NLS dataset can be seen in **Table 7**, with Massahusetts dataset in Table 8, and with Ottawa dataset in Table 9. Generally, the results were very similar to the results of training with a single encoder and color space. Furthermore, additional color space and encoder increases the number of parameters in the model. Considering the number of parameters and the results, adding different color space as an additional input doesn't improve the performance of the model.

	UNet	DenseUNet	RoadVecNet
RGB + HSV	69	69,4	70,7
RGB + LAB	69	69,6	71
RGB + YCbCr	69,4	69,5	71,1
RGB + YUV	49	69,7	70,8
HSV + LAB	68,3	69,5	70,6
HSV + YCbCr	68,7	69,5	70,4
HSV + YUV	49	69,6	70,1
LAB + YCbCr	68,7	70	70,3
LAB + YUV	49	69,5	70,5
YCbCr + YUV	49	69,5	70,2

Table 7: F1-scores of UNet, DenseU-Net, and RoadVecNet in different color space pairs with NLS dataset.

	UNet	DenseUNet	RoadVecNet
RGB + HSV	79,3	80,4	78,6
RGB + LAB	76,6	80,6	77,7
RGB + YCbCr	78,7	80,8	77,9
RGB + YUV	77,9	81,6	78,8
HSV + LAB	49,1	76,9	80
HSV+YCbCr	77,2	77,5	80,2
HSV + YUV	78,3	76,8	77,3
LAB+YCbCr	77,6	77,2	77,4
LAB + YUV	76,5	78,9	78,4
YCbCr + YUV	77,9	77,3	77,5

Table 8: F1-scores of UNet, DenseU-Net, and RoadVecNet in different color space pairs with Massachusetts dataset.

	UNet	DenseUNet	RoadVecNet
RGB + HSV	94,5	96,1	94,6
RGB + LAB	93,7	95,5	94,4
RGB + YCbCr	93,3	94,5	95,5
RGB + YUV	82,8	95,5	95,4
HSV + LAB	75,7	95,6	95,7
HSV + YCbCr	94,4	95,9	94,1
HSV + YUV	94,3	95,6	96
LAB + YCbCr	81,5	96	95,3
LAB + YUV	93,6	95,6	93,2
YCbCr + YUV	94	95,8	95

Table 9: F1-scores of UNet, DenseU-Net, and RoadVecNet in different color space pairs with Ottawa dataset.

RoadVecNet and DenseU-Net performed similarly with Ottawa dataset, while UNet performed slightly worse. With NLS dataset, however, RoadVecNet had the best performance and UNet the worst.

DenseU-Net had little variance in performance with different color space pairs with Ottawa dataset. Worst F1-score was with RGB + YCbCr color space pair, and best F1-score was with RGB + HSV. With NLS dataset, DenseU-Net had best

performance with LAB + YCbCr color space pair and similar performance with the other color space pairs.
RoadVecNet had worst F1-score with LAB + YUV color space pair, and best with HSV + YUV with Ottawa dataset, and worst F1-score with HSV + YUV color space pair and best with RGB + YCbCr color space pair with NLS dataset.
UNet got stuck in local minima with three color spaces on Ottawa dataset, with four color spaces on NLS dataset, and with one color space on Massachusetts dataset. The best F1-score was achieved with RGB + HSV color space pair with Ottawa dataset, and RGB + YCbCr color space with NLS dataset, and RGB + YUV with Massachusetts dataset. Additional input with different color space improved the performance of DenseU-Net the most.

5.4 Multiple outputs

The results of UNet, DenseU-Net, and RoadVecNet trained in different color spaces without sharpening or edge enhancement, with sharpening, and with edge enhancement with NLS dataset can be seen in **Table 10**, with Massachusetts dataset in Table 11, and with Ottawa dataset in Table 12.

	UNet	DenseUNet	RoadVecNet
Surface	69,3	69,8	71,54
Edge	55,2	55,2	55,57

Table 10: Road segmentation and road outline segmentation results from UNet, DenseU-Net, and RoadVecNet with two outputs in RGB color space with NLS dataset with F1-score

	UNet	DenseUNet	RoadVecNet
Surface	78,46	78,45	80,76
Edge	61,4	61,25	68,85

Table 11: Road segmentation and road outline segmentation results from UNet, DenseU-Net, and RoadVecNet with two outputs in RGB color space with Massachusetts dataset with F1-score

	UNet	DenseUNet	RoadVecNet
Surface	95	96,7	96,8
Edge	69,8	66,2	70,9

Table 12: Road segmentation and road outline segmentation results from UNet, DenseU-Net, and RoadVecNet with two outputs in RGB color space with Ottawa dataset with F1-score

Generally, road outline segmentation as an additional output improved the results compared to single-output road segmentation. However, additional encoder-decoder pair also increases the number of parameters of the model.
RoadVecNet has the best performance with each dataset, while UNet has the worst performance. Road outline segmentation had worse F1-score with each model compared to their road segmentation F1-score. This is partly due to road outlines being much smaller compared to the road surface and thus having less example pixels for training, and road outlines are more likely to have errors in labels because it can be hard to determine where exactly the road edge is in the pixels.

6. Conclusions

In this paper, the effects of color space and multi-task learning methods were studied for road segmentation from aerial image data using UNet, DenseU-Net, and RoadVecNet architectures. Multi-task learning methods involved augmenting models with additional encoders and inputs as well as incorporating smaller encoder-decoder pairs for road outline segmentation. The experiments were conducted with NLS orthophotos, Massachusetts road imagery dataset, and Ottawa dataset.
This study builds upon our previous research by evaluating the effectiveness of different color spaces in road segmentation tasks using additional datasets, sharpening and edge enhancement. While our earlier work indicated minor improvements when using certain color spaces, the current findings suggest that such differences are minimal when broader datasets are employed.
The results show that different color space had little effect on the performance of any of the neural networks with different datasets. RGB color space is a good choice for both road surface segmentation and road outline segmentation. Deep learning methods can adapt to different color spaces well and find similar features from each color space.
Using additional encoder and an additional input with different color space had little effect on the results, while also increasing the number of parameters of the network. Adding a smaller encoder-decoder pair to the end of the network with road outline segmentation task as an additional output increased the performance of the model slightly, but also increased the number of parameters of the network.
Furthermore, the experiments with sharpening and edge enhancement techniques did not significantly improve results. Performance comparison between the models indicated that DenseU-Net and RoadVecNet outperformed UNet in road segmentation tasks. Moreover, models achieved higher F1-scores on the Ottawa dataset compared to the NLS dataset, attributed partly to occlusions and labelling errors in the latter. Notably, RoadVecNet demonstrated superior performance on the NLS dataset, suggesting its efficacy in handling occlusions, possibly due to its DDSPP module's larger receptive field.
Lastly, the observed variations could be due to the inherent randomness in neural network training, which can cause slight fluctuations in performance outcomes. Thus, considerations such as dataset diversity, neural network architecture, loss functions, and augmentation techniques remain pivotal in enhancing neural network performance for road segmentation.

Acknowledgements

The work was supported by the National Land Survey's AI4TDB project (Artificial Intelligence for Topographic Database Accuracy Enhancement), which was funded by the Ministry of Agriculture and Forestry in Finland for the period of 1.1.2023-31.12.2023.

The authors also wish to thank CSC - IT Center for Science, Finland (urn:nbn:fi:research-infras-2016072531) and the Open Geospatial Information Infrastructure for Research (Geoportti, urn:nbn:fi:research-infras-2016072513) for computational resources and support.

References

Abdollahi, A., Pradhan, B., & Alamri, A. (2021). RoadVecNet: a new approach for simultaneous road network segmentation and vectorization from aerial and google earth imagery in a complex urban set-up. GIScience & Remote Sensing, 58(7), 1151-1174.

Bastani, F., He, S., Abbar, S., Alizadeh, M., Balakrishnan, H., Chawla, S., ... & DeWitt, D. (2018). Roadtracer: Automatic extraction of road networks from aerial images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4720-4728).

Caruana, R. (1997). Multitask learning. Machine learning, 28(1), 41-75.

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.

Dong, R., Pan, X., & Li, F. (2019). DenseU-net-based semantic segmentation of small objects in urban remote sensing images. IEEE Access, 7, 65347-65356.

Fukushima, K., & Miyake, S. (1982). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In Competition and cooperation in neural nets (pp. 267-285). Springer, Berlin, Heidelberg.

Henry, C., Fraundorfer, F., & Vig, E. (2021, January). Aerial Road Segmentation in the Presence of Topological Label Noise. In 2020 25th International Conference on Pattern Recognition (ICPR) (pp. 2336-2343). IEEE.

Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7132-7141).

Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning (pp. 448-456). PMLR.

Jha, A., Kumar, A., Pande, S., Banerjee, B., & Chaudhuri, S. (2020, October). MT-UNET: A Novel U-Net Based Multi-Task Architecture For Visual Scene Understanding. In 2020 IEEE International Conference on Image Processing (ICIP) (pp. 2191-2195). IEEE.

Jurio, A., Pagola, M., Galar, M., Lopez-Molina, C., & Paternain, D. (2010, June). A comparison study of different color spaces in clustering based image segmentation. In International conference on information processing and management of uncertainty in knowledge-based systems (pp. 532-541). Springer, Berlin, Heidelberg.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25, 1097-1105.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., & Jackel, L. (1989). Handwritten digit recognition with a back-propagation network. Advances in neural information processing systems, 2.

Liu, Y., Yao, J., Lu, X., Xia, M., Wang, X., & Liu, Y. (2018). RoadNet: Learning to comprehensively analyze road networks in complex urban scenes from high-resolution remotely sensed images. IEEE Transactions on Geoscience and Remote Sensing, 57(4), 2043-2056.

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).

Mnih, V. (2013). Machine learning for aerial image labeling. University of Toronto (Canada). (Pp. 84-90)

National Land Survey of Finland, 2021. “NLS Aerial photographs”. https://www.maanmittauslaitos.fi/en/e-services/open-data-file-download-service (6 December 2021)

Ozturk, O., Isik, M. S., Kada, M., & Seker, D. Z. (2023). Improving Road Segmentation by Combining Satellite Images and LiDAR Data with a Feature-Wise Fusion Strategy. Applied Sciences, 13(10), 6161.

Raninen, J. (2022). The Effect of Colour Space in Deep Multitask Learning Neural Networks for Road Segmentation (Master's thesis, Itä-Suomen yliopisto).

Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.

Saito, S., Yamashita, T., & Aoki, Y. (2016). Multiple object extraction from aerial imagery with convolutional neural networks. Electronic Imaging, 2016(10), 1-9.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Taylor, L., & Nitschke, G. (2018, November). Improving deep learning with generic data augmentation. In 2018 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 1542-1547). IEEE.

Xu, Z., Liu, Y., Gan, L., Sun, Y., Wu, X., Liu, M., & Wang, L. (2022). Rngdet: Road network graph detection by transformer in aerial images. IEEE Transactions on Geoscience and Remote Sensing, 60, 1-12.

Yang, M., Yu, K., Zhang, C., Li, Z., & Yang, K. (2018). Denseaspp for semantic segmentation in street scenes. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3684-3692).

Väylävirasto, 2021. “Digiroad, Kansallinen tie- ja katuverkon tietojärjestelmä”. https://vayla.fi/vaylista/aineistot/digiroad (6 September 2021).