Machine Learning Project 2: Road Segmentation

Team members: Jin Yan; Rongchen Wang
College of Management, EPF Lausanne, Switzerland

***Abstract*—This project aimed to implement a machine learning system to tackle a road segmentation task in which roads are extracted from the satellite maps. U-nets have been selected and schemes to advance performance such as data augmentation have also been utilized. Finally, a 0.885 F1-score and 0.940 accuracy has been achieved on the AICrowd test set.**

I. Introduction

Road segmentation is an important task in the field of computer vision and image processing. There are various methods that have been proposed for road segmentation. In this road segmentation project, we have extracted roads from the satellite maps. We have adopted several models such as U-nets (Baseline) and modified models based on U-nets as the inner structure of our neural network.

II. Data Pre-Processing and Analysis

A. Introduction to dataset

The training dataset contains 100 aerial images each of which has a size of 400x400 pixels and 3 channels. The ground truth of the traing data each also has the same pixel size with only one channel. Each pixels has the value of 1-0, indicating the road or background category pixel belongs to. The test dataset consists of 50 aerial images each of which has the same size as the training set, but each channel of which is of the size 608x608 RGB.

B. Data Augmentations

It's difficult to train a qualified model with small dataset so data augmentation methods are adopted for enlarging the training dataset. With a limiting dataset, models are easy to fall into traps of overfitting. Introducing modified or augmented versions of the original training images could better train the model in extracting and learning features in a way that is invariant to their position, scale, angle and so on.

1) Morphological Transformations: Applying morphological transformations can boost robustness of model against biases present in the original training data. Several morphological transformations have been applied to training data:
   - Random Rotation. Most of the images only have vertical or horizontal roads, therefore it may be difficult for models to recognize roads at angles. We rotated the training images by certain angle that is randomly chosen from 0 to 90. By applying such rotation, model can be trained with images with different angles.
   - Flip. This approach is aimed to flip the training images horizontally and vertically.
2) Color Transformations: This approach is to alter the colors, which changes certain color of image to another. We modified the hue, saturation, lightness (HSV) qualities to random numbers.
3) Scale: This is aimed to scale the images by zooming in or out the generated information in the images. Since the size has changed, scaled images have been cropped into the size of original 400x400 pixels.
4) Image Sharpen: We used laplacian operator to enhance the definition of edges and fine details in images. The Laplacian operator we used in this project is $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$. In image convolution, the kernel is centered on each pixel in turn, and the pixel value is replaced by the sum of the kernel mutipled by the image values.
5) Other Transformations
   There could be other transformations applied to training data including blurring, adding noise, changing contrast, or luminosity. However, the test and train data are images with consistent clarity and high resolution, therefore other transformations seems unnecessary.

III. Models and Methods

Fully Convolutional Network structure has been applied widely in segmentation problems. As firstly designed for biomedical image segmentation, U-Net has shown a great accuracy on segmentation problems.

A. U-Net

U-Net[1] is a deep learning-based method for semantic segmentation, which was proposed by Olaf Ronneberger et al. in 2015. It is a convolutional neural network (CNN) architecture, which consists of an encoder and decoder. The encoder downsamples the input image, and extracts high-level features. The decoder upsamples the enconded features and outputs the final segmentation map. The systematic structure of U-Net is shown in Figure 1.

Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations. The left part of U is feature extraction part. Encoder is used to extracted the feature of images and has four downsampling. In each step, the three-dimensional matrix of the image acted as input to two 3x3 convolutional layers with an activation function follows. Then, there is a

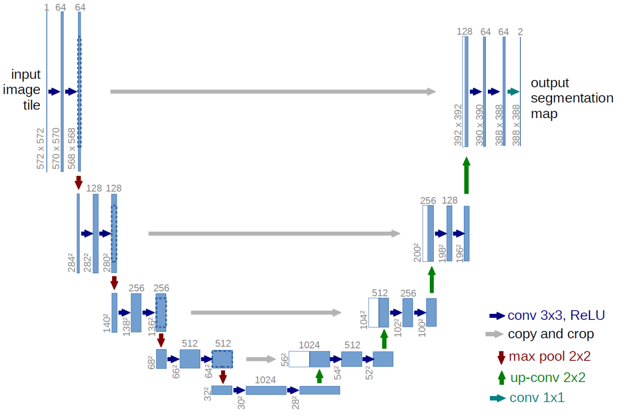


Fig. 1. U-Net Structure

maxpooling layer following each contraction step. During this process, the size of images is reduced while the number of channels increases, thus low-level features of input images are obtained by the networks.

Upsampling part (Decoder) extends the picture's height and width. Upsampling part has four upsampling steps corresponding to the encoder stucture. In each upsampling process, the matrix goes through 2x2 up-convonlutional layers firstly in which the size of the image is multiplied by 2, then two convolution kernels(conv 3x3 with an activation function follows). As a result of this reverse process of the encoder part, the size of images are extended and the number of channels decreases by half. After this process, combination with low-level features in encoder part is obtained.

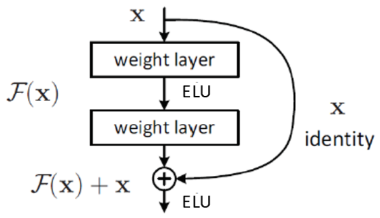
There are also four copy and crop parts also called skip connect, corresponding to each upsampling and downsampling. This part is aimed to generate information from both downsampling and upsampling processes. The network concatenates the low-level information and general information. If they are of different sizes or have different number of channels, the network crops them in order to complete concatenating.

B. Residual U-Net (Res-UNet)

The residual U-Net is a variant of the U-Net that uses residual connections in its architecture. A residual connection is a type of shortcut connection that allows the model to "skip" one or more layers, making it easier for the network to learn and improve performance. We adopted Res-UNet as an encoder to subsitute for the structure of U-Net. The basic block of Res-UNet is shown in figure 2. On the one side, the input goes through two 3x3 convolutions, on the other side, it is directly connected with the former one. The combination will go through the activation function ELU. Another difference is the size of each layer is half of that in U-Net.

C. Ringed Residual U-Net (RRU-Net)

Ringed Residual U-Net (RRU-Net) is a variant of the U-Net architecture, which is a convolutional neural network (CNN) for semantic segmentation. The RRU-Net was proposed by Guoyuan Wei et al. in 2019[2], and is designed to improve the performance of U-Net on the task of cell segmentation in microscopy images.



x
weight layer
ELU
weight layer
$\mathcal{F}(\mathbf{x})$
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
ELU

Fig. 2. Block of Residual U-Net

The RRU-Net architecture is similar to the U-Net architecture, consisting of an encoder and decoder. However, the RRU-Net introduces additional ring-shaped residual blocks in the encoder and decoder, which are designed to capture long-range dependencies in the input image and improve segmentation performance.

1) Residual Propagation
For solving the gradient degradation problem brought by contraction, the residual propagation to each stacked layers is introduced. The output y_f of the building block is defined as 1:

$$y_f = F(x, W_i) + W_s * x \tag{1}$$

where, x and y_f are the input and output of the building block, W_i represents the weights of layer i, the function $F(x, W_i)$ represents the residual mapping to be learned.

2) Residual Feedback
For further strengthening the differences of image essence attributes, the residual feedback is proposed, which is an automatic learning method and not just focus on one or several specific image attributes. Assume y_p is the output of residual feedback (input of the next iteration). y_p can be calculated as 2:

$$y_p = (sigmoid(G(y_f)) + 1) * x \tag{2}$$

where G is the linear projection. The network architecture of RRU-Net is shown in Figure3.

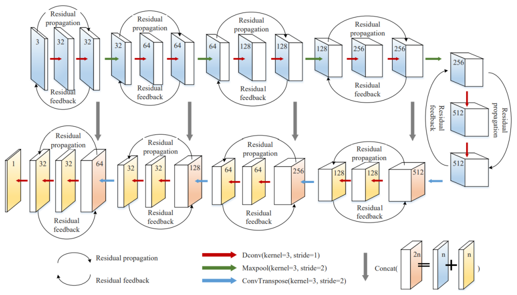


Fig. 3. The network architecture of RRU-Net

D. CBAM attention unit

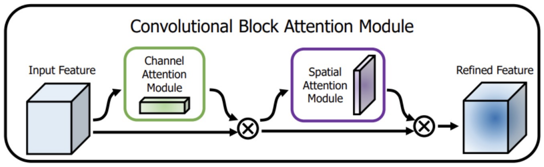


Fig. 4. The network architecture of CBAM

Convolutional Block Attention Module (CBAM)[3] is composed of two sub-modules called the Channel Attention Module (CAM) and the Spatial Attention Module (SAM). The channel attention generally want to figure out the meaningful part of the input image, and the spatial attention focuses on the location of this meaningful part.

Firstly, it passes the channel attention module where contains Max pooling and Avg pooling and then goes through a shared network adding sigmoid function. After that it passes the spatial attention module. In this module, we first use Max pooling and Avg pooling and concatenate them in order to describe the features more efficiently. These are convolved by a standard convolution layer.

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \tag{3}$$

$$M_s(F) = \sigma(f([AvgPool(F); MaxPool(F)])) \tag{4}$$

where c denotes channel attention, MLP represents multi-layer perceptron, s denotes spatial attention and f denotes a convolution operation.

In our experiment, we added CBAM in the last layer of Residual U-Net. For parameters, we choose reduction ratio = 4 and kernel size = 3.

E. Loss Function

Loss function plays an important role in training networks as it measures how well the model performs on the training dataset. Intersection over Union (IoU) loss function which is also called Jaccard Loss has been introduced. Jaccard Loss (IoU) could be described as equation 5,

$$IoU = \frac{P \cap T}{P \cup T} \tag{5}$$

Where P denoting the prediction, T denoting the target. IoU loss could increase the gradient of samples with high value of IoU and decrease the gradient with low value of IoU, therefore, the accuracy of the networks could be increased. Also, IoU could scale invariant as it is a ratio value. In order to avoid the gradient vanishing or exploding, a smooth term is introduced to the former equation 5. As a result, the final loss function is calculated as equation 6:

$$Loss = 1 - \frac{smooth + P * T}{smooth + P + T - P * T} \tag{6}$$

F. Model Development

At first, we used U-Net as the baseline for this road segmentation task, however, we found an obvious difference on the performance on training set and validation set. There may be several reasons behind the results achieved using U-Net: (1) The model might be over-fitting on the training set; (2) The data set is not large enough for training a complex model; (3) The parameter for the model is not well enough. As a result, we added a dropout layer to reduce the possibility of over-fitting; Moreover, we enlarge the training dataset by introducing more augmentation versions of original images. The training dataset would have 810 images.
Compared to U-Net, the residual connections introduced by Residual U-Net could help solve the degradation problem caused by increasing numbers of layers in the neural network. Residual U-Net helped increase the F1 score and accuracy. Then we added CBAM attention unit after Residual U-Net to expect that we can increase the feature power, but it seems like some overfitting problem occured or the receptive field was already sufficient, resulting in the decrease of accuracy. To further increase the performance of networks, Ringed Residual U-Net was introduced. Based on the structure of residual U-Net, a back convolution in the down-sampling layer and two Ringed Residual module in the up-sampling layer for residual feedback were added so that the model could better make use of the contextual spatial information in the images and help reduce prediction error.

IV. Evaluation Metrics

As area of the two classes - the road class and background class is not balanced, therefore, F1-score could be a better metric than accuracy to measure the performance of models.

$$F1 - score = \frac{2}{recall^{-1} + precision^{-1}} \tag{7}$$

F1-score is calculated based on equation 7. Precision is the fraction of real road pixels among the model-detected pixels. Recall is the fraction of the total amount of road pixels that were actually detected. As F1-score takes both false negative and positive values into consideration, it has been used as metric in the project for evaluation.

V. Model Selection

A. Learning rate

Learning rate can have a huge influence on the convergence of the model. When learning rate is too large, the loss may get fluctuated while the small one may cause the model converge slowly. So we tried to tune the parameter one by one, namely $3 * 10^{-4}$, $4 * 10^{-4}$, $5 * 10^{-4}$ and $7 * 10^{-4}$ to see which one could have best performance.

B. Batch size

Considering the computational power and the nature of road segmentation task, we set batch size equaling to 6.

C. Activation Function

Compared to ReLU function, ELU has some advantages which could overcomes the dying ReLU problem. Also, the function tends to converge cost to zero faster and produces more accurate results. As ELU is more of a merger between good features of ReLU and Leaky ReLU, so in this project, ELU was used as the activation function and helped increase the F1 score and accuracy.

D. Dropout

Dropout has been added after activation function to act as a common regulation way. Large dropout rate may cause a slow convergence and set the dropout rate to 0.2 would help the model generalize well while avoiding the slow convergence.

E. Eliminating Noise from prediction

Based on the observation of prediction on test images, some area similar to the road class would mislead the model, which would cause small blurs on the predictions in figure 5. By using image morphological transformations, these noise was eliminated from the predictions, thus increasing the accuracy. Use $skimage.morphology.remove\_small\_objects$ to help remove the noise in the prediction. We made use of the difference of area of pixels between narrow roads and noise through setting the threshold pixel area as 800 and remove the small objects below this area.

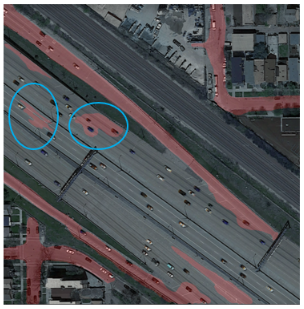


Fig. 5. Noise in the prediction

VI. Results

We uploaded our results to AIcrowd, below is the metrics that we obtained from the website:

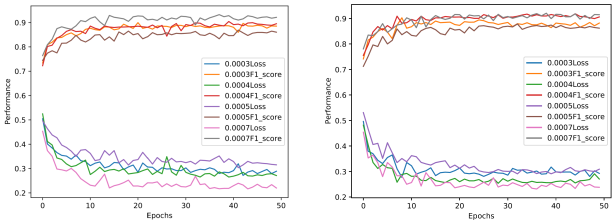
TABLE I
Model performance with other parameters maximized

Network	Accuracy	F1 Score
U-Net (Baseline)	0.932	0.874
Res-UNet	0.938	0.885
Res-UNet(CBAM)	0.938	0.882
RRU-Net	**0.940**	**0.885**

Our baseline is U-Net. But we fould it is time-consuming. It need to run about 3 hours while others only take about 1 hour. The best result we got is RRU-Net with final tuned parameters shown in Table II. From Table I, we noticed that Res-UNet and RRU-Net is clearly better than the simple U-Net. However, when we added CBAM attention unit after Res-UNet, there is a slightly decrease in Accuracy and F1 Score. There might be overfitting problem.

TABLE II
Final Tuned Hyperparameters of RRU-Net

Parameters	Value
Learning rate	$7 * 10^{-4}$
Batch size	6 (considering computational power)
Activation function	ELU function
Dropout	0.2 considering it increasing the generality of model
Loss Smooth term	20



(a) loss of RRU-Net with different learning rate (b) loss of Res-UNet with different learning rate

Fig. 6. loss of RRU-Net and Res-UNet

We have trained different models and compared the performance of models with different parameters, comparisons between different learning rate for Res-UNet and RRU-Net is shown in Figure 6. Then we can find that when increasing the learning rate to $7 * 10^{-4}$, the performances become better for both models.

VII. Summary and Outlook

Further work can be done to improve the model: Firstly, if computational power is permitted, training dataset could be enlarged; Secondly, the hyper-parameters could be better tuned with appropriate cross validation; Thirdly, we didn't solve the overfitting problems with the Res-UNet(CBAM), however, trying different parameters actually increases the accuracy and performance of models.

References

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[2] X. Bi, Y. Wei, B. Xiao, and W. Li, "Rru-net: The ringed residual u-net for image splicing forgery detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019, pp. 30–39.

[3] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.