Semantic segmentation of TRIMBLE dataset

BRAHAMI Sara SARA.BRAHAMI6@GMAIL.COM

Paris, France

Editor: Machine Learning Avancé (2021-2022)

Abstract

In this paper we introduce, a deep learning model based on U-net `https://arxiv.org/abs/1505.04597`, for semantic segmentation of class Road and field. The U-Net architecture is a popular choice for image segmentation tasks, particularly in the field of biomedical image analysis, due to several advantages :

Effective use of limited data : The U-Net architecture is particularly useful when the available training data is limited. This is because the architecture includes skip connections that allow the model to reuse features from the contracting path during the expansive path, which can help to improve segmentation accuracy even with a limited amount of data.

High accuracy : The U-Net architecture has shown high accuracy in many image segmentation tasks, particularly in the biomedical domain, where it has been used to segment organs, nuclei, and cells in microscopy and medical images. The architecture's ability to capture both local and global contextual information and the use of skip connections to preserve low-level information and details can help improve accuracy.

Flexibility : The U-Net architecture can be modified to suit different types of image segmentation tasks. For example, researchers have adapted the U-Net architecture for multi-class segmentation, 3D segmentation, and even for non-image data like speech signals.

Easy to implement and train : The U-Net architecture is relatively simple and can be implemented and trained efficiently. This is particularly useful in applications where real-time processing is required, such as in medical image analysis or autonomous driving.

In summary, the U-Net architecture is a popular choice for image segmentation tasks due to its effectiveness, high accuracy, flexibility, and ease of implementation and training.

Our model takes a 3-dimensional image as input and we get in output a segmented image containing labeled pixels.

Keywords: U-Net

1. Database

The dataset contains 107 images of the road class and 45 images of the field class, our goal is to do the supervised learning, so to do it we need the annotations, for that I labeled the database using MATLAB's image labeler software. For each image, there is an associated PNG file with a mask. The size of a mask equals to the size of the related image. Each pixel in a mask image can take one of three values : 0 for unlabeled, 1 for road class and 2 for field class.

Images are large and they have a different sizes, to improve training and generalization, we can apply image augmentation. This procedure generates images with random distortions or modifications. The torchvision.transforms library makes this process easy to apply simultaneously for image and mask. Here, transformations are defined for the training set and just resizing and normalization for the validation set :

2. Experience

2.1 Training

As an **optimizer**, i choose Adam because he adjusts the learning rate of each weight parameter during training based on the estimated first and second moments of the gradients. This means that the learning rate can be adapted on a per-parameter basis, allowing the optimizer to converge quickly and efficiently and it require very little memory compared to other optimizers like stochastic gradient descent with momentum (SGDM). This makes it a suitable choice for training larger models or when working with limited computational resources. with an initial learning rate of 0.00001 and batch size of 64. I divided our dataset randomly to 95 % for training and 5 % for validation.
Loss : crossEntropyLoss for multi classes was applied, it is often used as the loss function for neural networks that are trained to classify data into multiple classes, defined as :

$$Loss = \frac{-1}{N} \sum_{i} (yi * log(pi)) \tag{1}$$

where : N is the number of samples in the training data, yi is the true class label of the ith sample, pi is the predicted probability of the ith sample belonging to the positive class.

Evaluation metrics : Dice coefficient (also known as F1 score) is a commonly used metric for evaluating the performance of segmentation models. The Dice coefficient measures the overlap between the predicted segmentation and the ground truth segmentation of an image.

The Dice coefficient is defined as :

$$Dice = \frac{2 * TP}{2 * TP + FP + FN} \tag{2}$$

2.2 Experimental Results

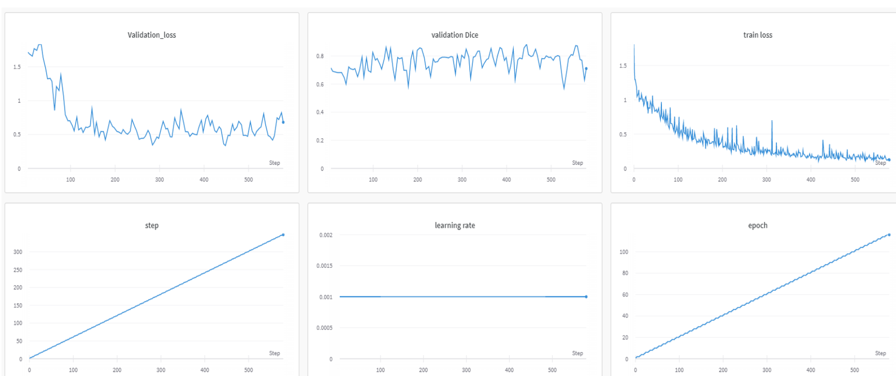

FIGURE 1 – figures of taring and validation dataset

As we can see from the figures above, our model is able to learn the images. This is represented by the decrease in the Loss errors during training and validation. The curve on the figure also shows that the model can further improve its accuracy by training on more epochs.

2.3 Discussion

We have a good precision 3 of the road class : 80 % because we have a lot of images which contain road compared to the field which has 20 % To solve this, we need to increase the data of

FIGURE 2 – Ground Truth, color red represent road and blue field

FIGURE 3 – Predict mask, color red represent road and blue field

images containing field, and it depends on the pixels, we need to have many images with green pixels, especially when the images have both road and field. Due to the short duration and availability of the clusters, the model did not train for a long time. Another option to solve the class imbalance problem is to divide the cross-entropy loss by the weight of each class.