Convolutional Neural Networks for Image Segmentation

Chibuzor John Amadi

[1] University of Pavia, Pavia
[2] University of Milano-Biccoca, Milan
[3] University of Milano-Statale, Milan

Abstract. Convolutional Neural Networks(CNNs), in recent times have been applied to many fascinating fields, such as image classification and audio classification. This report investigates the use of CNNs for semantic segmentation. For image segmentation, unlike classical CNN, we will require an encoder and decoder in classification which will be elaborated in section 2. The UNet will be the encoder, decoder for this project, the encoder will be part of the CNN for classification and the decoder will restore the original spatial resolution and reducing the number of features. For the purpose of this task accuracy will be optimal, regardless we will introduce the Intersection Over Union (IOU) as a performance metric.

Keywords: Semantic segmentation · UNet · IOU

1 Introduction

Semantic segmentation is a computer vision task in which the goal is to categorize each pixel in an image into a class or object. Segmentation has been applied in recent years to many domains such as medical imaging and autonomous driving etc,. A normal Convolutional Neural Neural for image classification will include two main components. The first being the convolutional layers for extract local features from the input image, here the number of features increases layer after layer. The pooling layers reduce the spatial resolution and the global pooling reduces the local features to a single vector. The second component is a multi-layer perceptron(MLP) made of fully connected layers, ending with a vector of probability estimates. Convolutional Neural Networks here used for segmentation will entail the removal of the MLP component in order to preserve spatial information. Instead, a decoder is introduced to restore the original resolution and perform a pixel-level classification. Semantic Segmentation[1] address the tension between semantic and location. In a typical classification tasks, where the output to an image is a single class label, with segmentation the desires output should include localization. The idea of the encoder, decoder was first seen in [2] with the fully convolutional Network(FCN). The model will include batch normalization and will use a cross entropy loss function. The breakdown of the UNet and the FCN will be in section 2 and the evaluation of the model will be shown in section 3.

2 Model Architecture

2.1 Fully Convolutional Network and U-Network

The U-Net 1 convolutional network was specifically designed for biomedical image segmentation tasks, which requires precise localization and context identification are crucial. The model distinguishes its symmetric encoder-decoder structure and makes use of skip connections[3], which facilitates the preservation of spatial data and features across the network, essential for detailed segmentation tasks. In the project the UNet [3] was built with the FCN[2].

The encoder of the U-Net functions as the feature extractor, which serves to condense information from the input image. The processing of the input image includes two consecutive convolutional layers, each with 64 filters of size 3x3, stride 1, and padding 1, ensuring the spatial dimensions are maintained. Each convolution is followed by a batch normalization layer, which normalizes the activation, helping to maintain stability and improve training. The output is then down-sampled by max pooling, reducing the resolution while doubling the depth of features. The number of filters is increased to 128, improving the U-Nets ability to capture more complex features. The structure mirrors the first stage with two sets of convolutions and batch normalization, followed by another down-sampling step. Further increasing the filters to 256, processes the data to extract even more detailed features before passing it on to the bottleneck.

The bottleneck is an important of the network, it is critical for transitioning between feature encoding and decoding. With two convolutional layers with 256 filters each, followed by batch normalization and ReLU activation, this deals with most compressed feature representation from the encoder. The bottleneck concludes with an 'Upsample' initiates the process of expanding the spatial dimensions of the feature maps. This is done using a nearest neighbor approach, which effectively doubles the resolution of the feature maps, setting the stage for the decoding process. The decoder reconstructs the spatial dimensions of the output from the condensed feature maps, refining the segmentation details.

Features from the bottleneck are combined with those from the corresponding encoder stage through skip connections, which help preserve edge details and local structures essential for precise segmentation. The decoder then applies two convolutional layers with 128 filters each, followed by upsampling to enhance spatial resolution. It then refines the features, using 64 filters and again employing convolutions and batch normalization. The spatial resolution is increased once more through upsampling, preparing for the final reconstruction of the image's original dimensions. The upsampled feature maps to adjust the channel depth back to 3, aligning with the typical RGB output for segmentation masks. Convolutional operations here fine-tune the details and prepare the data for the final output layer.

The segmentation map is produced by a final 2D convolutional layer with a kernel size of 1x1. This layer acts as a classifier at each pixel. Each convolutional layer (except in the output) is followed by a ReLU activation function to introduce non-linearity, enhancing the model's ability to learn complex patterns. The

final output layer typically uses a classifier that takes the number of classes of the model.

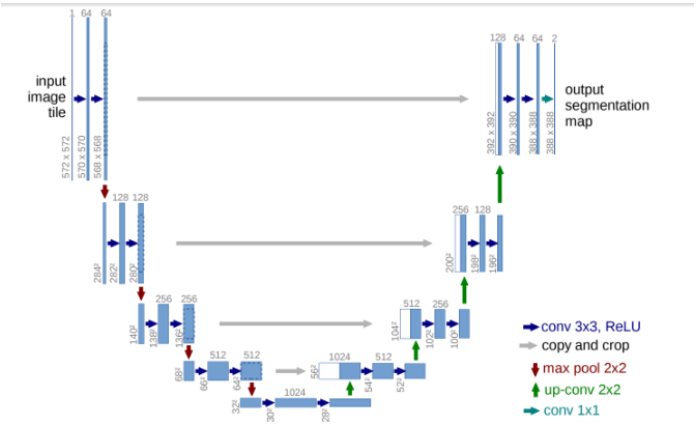

Fig. 1. U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Fig. 1. Diagram of the U-Net [3]

3 Evaluation

3.1 Experiment details

For the purpose of this project, the Oxford Pet dataset [4] was used. The dataset contained both original and segmented pictures of cats and dogs, a total of 7349 images. The segmented masks were carried out manually. The dataset is distributed into train and test folders, each folder includes 37 folders representing different breeds of dogs and cats. It is important to note that the breeds of these animals are in no way relevant to the task.

For training this model, the Adam [5] optimizer was utilized and the loss function was the cross entropy loss function. The accuracy and the Intersection Over Union(IOU) were the performance metrics used on the model. The IOU is very important in semantic segmentation as it reflects the classification performance of each class of pixels in the image. We can defined the accuracy of the model by plotting and analysing the confusion matrix of the model's performance on the test set. equation 1 defines the accuracy of the model with relation to the confusion matrix of the model. We sum the diagonal of the confusion matrix

which represents the true positives of the classes and we divide by the total sum of the confusion matrix.

$$Accuracy = sum(TPofeachclass)/Totalsumofsamples \tag{1}$$

$$Accuracy = sum(diagonalCM)/Totalsum(CM) \tag{2}$$

$TP = True\ Positive,\ CM = Confusion\ Matrix$

The IOU, shown in Fig 2 will be defined by dividing the number of true positive predictions for each class (diagonal of the confusion matrix) by the union of the predicted and actual values for that class (sum of the corresponding row and column, minus the diagonal element for that class). The mean IOU is then the average of these values across all classes.

$$IOUclass0 = TP, 0/(sumP, 0 + sumA, 0) - TP, 0 \tag{3}$$

$TP = True\ Positive,\ P = Predicted\ class,\ A = Actual\ class$

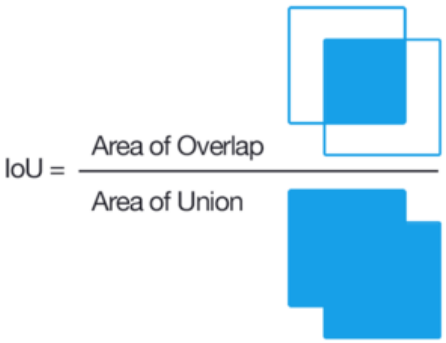

Fig. 2. IOU performance matrix

4 Results Analysis and Conclusion

The analysis derived from training the model with the adam optimizer and cross-entropy loss function was on consistency 86 percent. The accuracy derived when the model was introduced to the test set reduced to a percent of 81 percent. The model is over-fitting and clearly indicates work needs to be done on generalization. The IOU derived by the model with the test set was on consistency 0.627. This result also can be improved by fine-tuning the model, further experiments on the loss function and optimizer, data augmentation can be carried out as well. Unfortunately, in this project, due to computational resources, numerous experiments will not be conducted. However, in this project, analysis was carried out to understand the IOU and the accuracy evaluation. Fig3 5 shows the

lowest IOU scored by any image in the classification. This image, just like many others with low IOU performance, can be categorized into images where the animals can not be clearly distinguished from the background due to the lack of a clear edge and in situations where the image of the animal dominates the entire picture. Rare cases include instances where the image of the dog looks like a cat or vice versa.

Comparing the IOU and accuracy metrics, accuracy is most likely to achieve higher results. The difference can come from the fact that IOU is more sensitive to each class than accuracy, which is a good reason to always consider both of them in semantic segmentation. The IOU on each class will give information about object or item overlaps. In some cases, the IOU will score low when the classes in an image are imbalanced. Another case is when the objects in the image are small or there is no precision between boundaries. Edge detection on images will be an optimal solution to improve the IOU of the classification. To conclude, from this project, we can see that convolutional neural networks with the U-Net is a very good technological option for dealing with semantic segmentation.

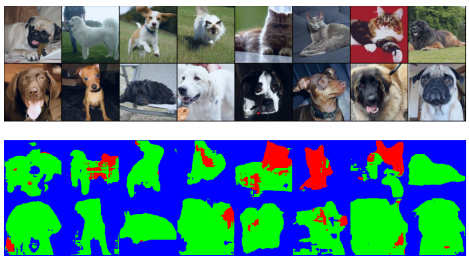

Fig. 3. Original Images and Test Segmentation Result

5 references

1 "Semantic/Image Segmentation" en.wikipedia.org/wiki/Image-segmentation
2 "Fully Convolutional Networks for Semantic Segmentation.". Jonathan Long, Evan Shelhamer,and Trevor Darrell, UC Berkeley,2015.
3 "U-Net: Convolutional Networks for Biomedical Image Segmentation" Olaf Ronneberger, Philipp Fischer, and Thomas Brox,2015
4 "The Oxford-IIIT Pet Dataset" Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman and C. V. Jawahar
5 "Stochastic Gradient Descent." Wikipedia, en.wikipedia.org/wiki/Stochastic-gradient-descent/Adam

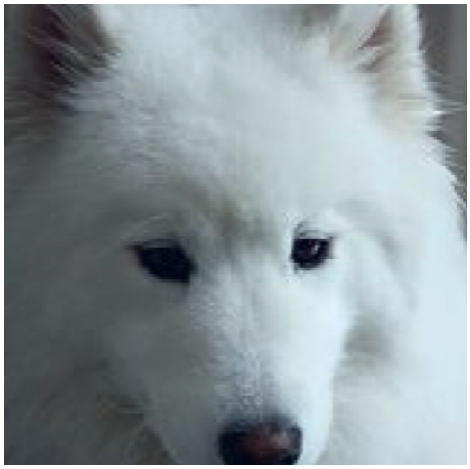

Fig. 4. Example of Class Imbalance image

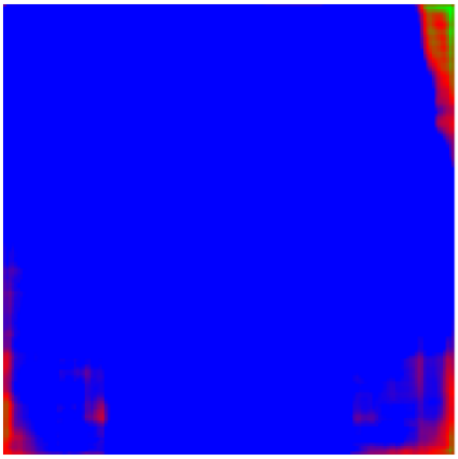

Fig. 5. Segmented image of Class imbalance image, IOU = 0.142