# Semantic Segmentation on Oxford Pet Dataset Using FCN

1st Sijia Hua
*Science Academy*
*University of Maryland*
College Park, United States
tomhuasj@umd.edu

***Abstract*—This project aims to present a semantic segmentation on a pet dataset using FCN algorithm. The primary objective is to identify the pixels for pets and background. The model is trained using PyTorch. In this paper, I will introduce my approach and detailed steps to reproduce the result.**

***Index Terms*—Oxford-IIIT Pet Dataset, Semantic Segmentation, Computer Vision**

## I. Introduction

Semantic Segmentation is crucial in applications such as autonomous driving system and image editing, where distinguishing object boundaries is essential. The goal of this project aims to implement and provide a semantic segmentation based on the training data from Oxford Pet dataset. The model mainly focuses on provide semantic segmentation for animals.

## II. Literature Survey

There are various amount of semantic segmentation methods have been proposed, and this project aims to reproduce the model from paper FCN, using U-Net, introduced by Long et al. The algorithm uses convolution neural network as the backbone for training. In the recent years, a lot of semantic segmentation task can also be performed using ViT (Vision Transformer) such as Segformer, which can be used in more complex scenarios, such as semantic segmentation over street views and cityscapes.

In my project, I have utilized a different neural network backbone for the FCN model training. Specifically, I chose to use ResNet50 as the backbone, the model which proved its success on ImageNet.

## III. Methodology

This section outlines the approach of semantic segmentation, the dataset, and the procedures used to develop and reproduce the result of the semantic segmentation using FCN.

### A. Tools and Environment

- **Dataset:** Oxford-IIIT Pet Dataset consists 37 different species of animals, along with around 200 images for each species. The annotated trimaps from the annotation specifies the labels for semantic segmentation task
- **Programming Language:** Python
- **Libraries and Tools:** Pytorch, Matplotlib

### B. Data Preprocessing

The images from the dataset contains several corrupted images, even from the original dataset, which prevents the data from loaded and transformed using cv2 and PyTorch. As a result, the first step is to identify and remove all the corrupted images from the dataset, and remove the corresponding trimap labels from the dataset.

After removing all the corrupted images, the next step is to prepare the data for the FCN model. In this project, I chose to use Resnet50 as my backbone model. All images are resized to 224x224 pixels in order to fit into the Resnet50 model, and normalized using same algorithm from PyTorch. The next step is to process the labels. Since I have transformed all the image size, which means I also need to transform the corresponding label values. All the pixels contain the pet borderline are labeled as 1, the other are labeled as 0.

### C. Model Architecture

In the beginning of the project, I have choose to use a simple CNN with only a few hidden layers for the model training. However, the results failed to capture the features and leads me to use a more complex model as the backbone for this project.

The model is based on ResNet-50 backbone, with a pretrained value to reduce the training time. The final layer of the model is modified to give binary output for the pet and background separately. The last layer is modified to apply convolution from 512 features to 2 labels as the results.

### D. Training Procedure

The dataset is randomly split into two sections: training data and test data, with a portion of 0.8 and 0.2 correspondingly. In my model, I chose to train the pretrained ResNet50 model using only 10 epochs, since To reproduce the result, the model uses the following parameters:

- Learning rate: 0.001
- Optimizer: Adam
- Loss Function: Cross Entropy Loss
- Epoch: 10

### E. Training Loss and Validation

By using a pretrained model, the model converges quickly and greatly reduce the time for training and provide a better

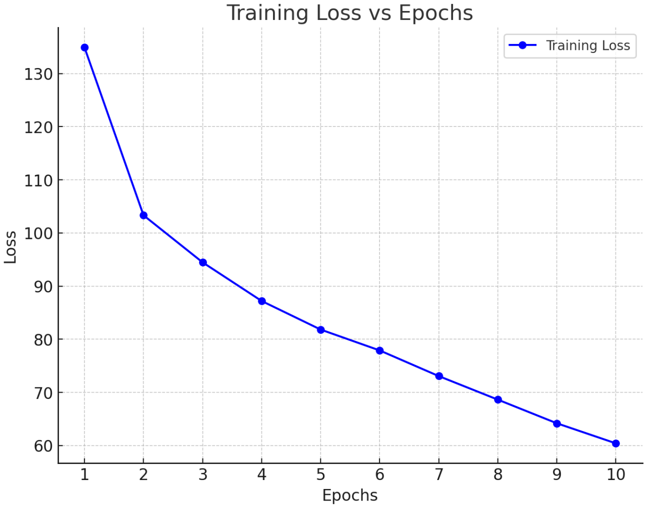

Fig. 1. Output 1

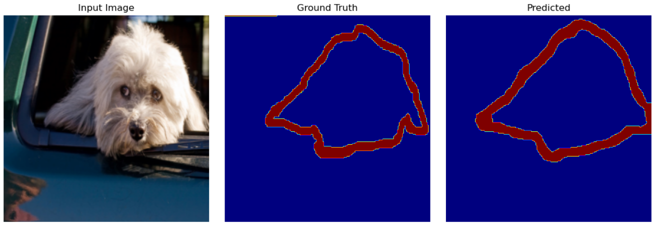

Fig. 2. Training Loss of 10 epochs

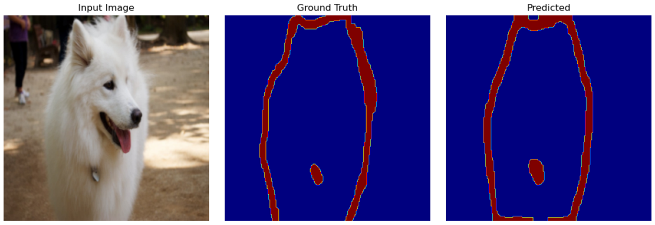

Fig. 3. Output 2

result. The Fig. 1 The model is mainly focuses on semantic segmentation, which makes it hard to provide a numeric value for evaluation. As a result, I choose to randomly sample images from the dataset and validate the data manually. The following figures Fig. 2, Fig. 3, Fig. 4 shows that the model is capable of finishing the semantic segmentation process based on the graph of the validation.

## IV. Results

The model's training loss decreases steadily over 10 epochs, with about 60 loss at the end of most training procedures. Beyond all the current results, I have also tested the model using different animals, including those species do not appeared in the training dataset.

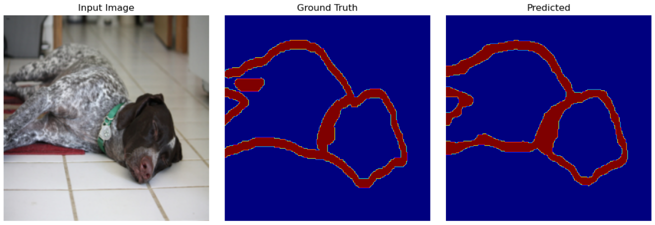

Fig. 4. Output 3

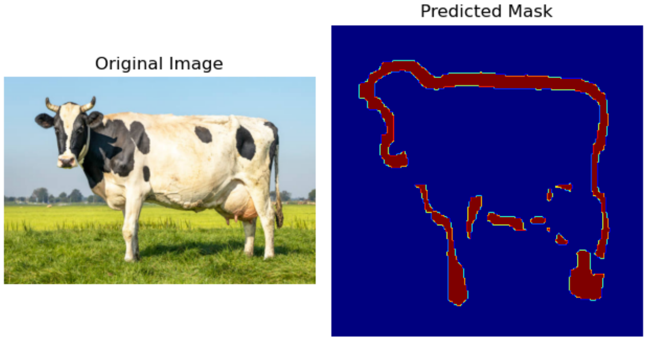

Fig. 5. Test result of a cow

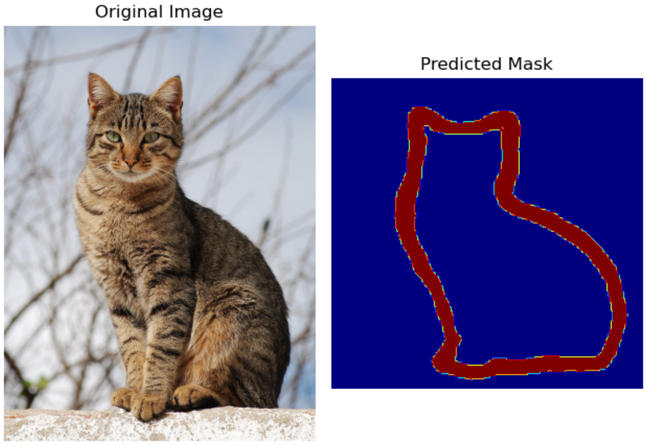

Fig. 6. Test result of a cat

The Fig. 5 shows the output result of a cow image, which is not represented in the dataset. However, the model shows that it is capable of capturing most features from the image, which shows the adaptability of the model.

The Fig. 6 shows another output using a random cat image from the internet. The model shows a clear outlines for the borderline of the cat.

## V. Discussion

From the results above, we can see that the FCN model performed well on a binary segmentation task, which is capable of identifing the regions of animals or pets in most cases. However, the training result may not be desirable when encountering animals beyond the dataset. From the cow figure, we can see that the borderline is not clear for the model, which means that there could be more improvements to the model.

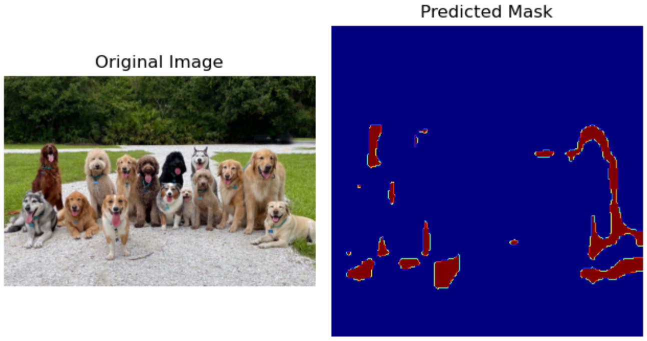

Fig. 7. Fail to detect the border

Another improvement for this project is to detect the animal type at the same time. However, the dataset do not contains a various range of species.

The original statement of this project aims to provide semantic segmentation over cityscapes, which includes multiple labels as the training result and utilize a vision transformer as the backbone for the project. However, due to the computing power and time limitation, I chose to train the semantic segmentation on a smaller dataset with binary classification.

The model also fails to provide correct semantic segmentation when various amount of pets are appeared in the picture at the same time. In Fig. 7 is an example of the failure.

## VI. Conclusion

In this project, the model successfully demonstrated its capability of semantic segmentation. By implementing the FCN model using Resnet50 as the backbone, the model successfully segment the pet regions and background, provide a consistent result after testing various times.

There are limitations for the model. The model lacks accuracy of showing animals which is not included in the dataset. One of the potential solutions is to increase the size of training dataset, and increase the number of species in the dataset. Another drawback of this project is that I am not able to use the output as feature to classify the species of the animals at the same time. Instead, using two different neural networks shows a better solution to such problem.

In the future work of this project, I would like to use the vision transformer as the backbone for training, and test the model's capability when face multiple animals as the input.

## VII. References

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation.

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., & Luo, P. (2021). SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers

University of Oxford, Visual Geometry Group. (n.d.). Oxford-IIIT Pet Dataset. Retrieved May 12, 2025, from https://www.robots.ox.ac.uk/ vgg/data/pets/