Camouflage Object Masking-WiDS'22

Mentored by Gaurav Misra
Anubha Vyasamudri, 210100019
January 29, 2023

Introduction

Camouflage Object Masking deals with the task of identifying and segmenting out objects that are not well-distinguished from their backgrounds. This is a specialized problem within the domain of Computer Vision, which has been seeing and continues to see rapid advances and novel techniques.
Camouflage Object Masking has several applications across varied fields including medical image segmentation, rare species discovery, the military, and so on. Recent work also suggests that research in AI-based segmentation of camouflaged objects could hold key insights into how the hierarchial levels of human visual perception are structured.

Weekly Learning

Week 1: Introduction to Machine Learning and Deep Learning-Andrew Ng's Deep Learning Specialization on Coursera; Introduction to PyTorch

Week 2: Convolutional Neural Networks(CNNs); Detailed learning of PyTorch; Some Reading on CNNs; MIT OCW Lecture on CNNs; course on CNNs on Coursera, by DeepLearning.AI

Week 3: Studied the Problem of Camouflage Object Masking; Read papers on Segmentation Network, Camouflage object masking and evaluation metrics for segmentation models

Week 4 and 5: Implemented the architectures to build a working model for desired results

Learning Outcomes

- Machine Learning and Deep Learning

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.Machine learning is an important component of the growing field of data science. Through the use of statistical methods, algorithms are

trained to make classifications or predictions, and to uncover key insights in data mining projects. These insights subsequently drive decision making within applications and businesses, ideally impacting key growth metrics. Machine learning algorithms are typically created using frameworks that accelerate solution development, such as TensorFlow and PyTorch.

Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to “learn” from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy. Deep learning drives many artificial intelligence (AI) applications and services that improve automation, performing analytical and physical tasks without human intervention.

- Convolutional Neural Networks

Convolutional neural networks are a specialized type of artificial neural networks that use a mathematical operation called convolution in place of general matrix multiplication in at least one of their layers. They are specifically designed to process pixel data and are used in image recognition and processing. A convolutional neural network consists of an input layer, hidden layers and an output layer. In any feed-forward neural network, any middle layers are called hidden because their inputs and outputs are masked by the activation function and final convolution. In a convolutional neural network, the hidden layers include layers that perform convolutions.

- Convolutional Networks for Image Segmentation

The U-Net architecture consists of a contracting path (left side) and an expansive path (right side). The contracting path follows the typical architecture of a convolutional network. It consists of the repeated application of two 3x3 convolutions (unpadded convolutions), each followed by a rectified linear unit (ReLU) and a 2x2 max pooling operation with stride 2 for downsampling. At each downsampling step we double the number of feature channels. Every step in the expansive path consists of an upsampling of the feature map followed by a 2x2 convolution (“up-convolution”) that halves the number of feature channels, a concatenation with the correspondingly cropped feature map from the contracting path, and two 3x3 convolutions, each followed by a ReLU. The cropping is necessary due to the loss of border pixels in every convolution. At the final layer a 1x1 convolution is used to map each 64-component feature vector to the desired number of classes. The U-Shape comes from the encoding and decoding blocks. It is a very common architecture for Image segmentation.

- Network Design

There is no guarantee that a camouflaged object is always present in the scene. Therefore, a method that systematically segments objects for any image will not work. Moreover, directly applying discriminative features from segmentation models (i.e., semantic segmentation and salient object segmentation, etc.) to camouflaged object segmentation is not effective because camouflaged objects conceal their texture into the surrounding environment. In order to segment camouflaged objects, we need an additional function that identifies whether a camouflaged object exists in an image. To do so, each pixel needs to be classified as a part of camouflaged objects or not. Such classification can be used not only to enhance segmentation accuracy but also to segment multiple camouflaged objects. Indeed, this classification only strengthens features extracted from the camouflaged part and weakens features extracted from a non-camouflaged part. Accordingly, segmentation and classification tasks should be closely combined in the network with different architectures for camouflaged object segmentation.

Description of the model

Choosing the Dataset
In our dataset, it is noteworthy that multiple objects, including separate single objects and spatially connected/overlapping objects, possibly exist in some images. This makes our dataset more challenging for camouflaged object segmentation. The challenge of the dataset is also enhanced due to some attributes, i.e. object appearance, background clutter, shape complexity, small object, object occlusion, and distraction:
• Object appearance: The object has a similar color appearance with the background, causing large ambiguity in segmentation.
• Background clutter: The background is not uniform but contains small-scale structures or is composed of several complex parts.
• Shape complexity: The object has complex boundaries such as thin parts and holes, which is usually the case for the legs of insects.
• Small object: The ratio between the camouflaged object area and the whole image area is smaller than 0.1.
• Object occlusion: The object is occluded, resulting in disconnected parts or in touching the image border.
• Distraction: The image contains distracted objects, resulting in losing attention to camouflaged objects. This makes camouflaged objects more difficult to discover even by human beings.

Uploading the Dataset:

The dataset is loaded onto the computer. There is a Test dataset, a validation dataset and the Train dataset. The U-Net architecture:We convert images into vectors. We have functions to create an encoder block, a convolutional block, and decoder block. These 3 blocks are what formed the basis of the UNet. The U-Net in the code consists of 3 encoder blocks, a convolutional block called the ”bottleneck” and 3 decoder blocks. We set the number of epochs and run through the training set. Displaying the Outputs: The outputs on the test dataset were displayed in a pre-defined location on the computer.

Results and Conclusions

I obtained an average dice score of 0.41 on this dataset. While the model worked well for some images, it also failed for a few other, so there is definitely scope for improvement in precision and accuracy, which can be kept in mind for my next project.