Readme

# Dataset:

The dataset contains 20 image-mask pairs of cars. Some of the images are in landscape orientation while others are in portrait. The corresponding masks are also oriented in the same manner.

The ground truth masks denote the car, i.e. `white` represents the car while `black` represents the background.

U2Net Repository: <https://github.com/xuebinqin/U-2-Net>

# Coding Setup:

You may use either PyTorch or Tensorflow.

Do any sort of Training on Google Colab and share with us the notebook along with a detailed description of each step and explain your reasons behind using a particular hyperparameter value, optimizer, loss function, augmentation etc.

# To be Shared:

* The colab notebook.
* The model weights.
* Inference script.

# Task 1:

* Train a U-2-Net model on this dataset after splitting it into training (15 images) and validation set (5 images).
* You may choose any optimizer, loss function and augmentation you deem appropriate.
* You are free to use any metrics, however, ‘dice\_score` and ‘IoU` must be among them.
* Keep model’s input\_size as 320x320.

# Task 2:

* Retrain the model (on the same training and validation dataset) with ‘dice\_loss` as the loss function.
* The metrics should include at least `dice\_score` and `IoU`.
* Keep model’s input\_size as 320x320.

# Task 3:

* Design a logic/algorithm/ML model to differentiate between Horizontal and Vertical cars (cars that have been rotated by 90 degree). The attached dataset contains such photos.
* Your proposed solution should be agnostic to the image resolution and aspect ratio.

# Task 4:

* Design a logic/algorithm to add a virtual background to the resulting car mask, while keeping the car grounded, for example, on a floor/white surface.
* Bonus: Simulate a realistic shadow on the bottom of the car.
* An example is shown below:



(Source: <https://www.pinterest.com/pin/850828554596972628/>)