

Deep Learning Challenge

You are provided dataset where the ID is fully visible in some images, partially visible and not visible in some images. Further, the camera you use to take the picture has noise, thus the images are noisy and corrupted. You are tasked to create a deep learning pipeline that classifies whether the image is visible, partly or not-visible at all. The csv file lists the .png file name and the corresponding ground truth label, such as

GICSD_1_0_13.png FULL_VISIBILITY

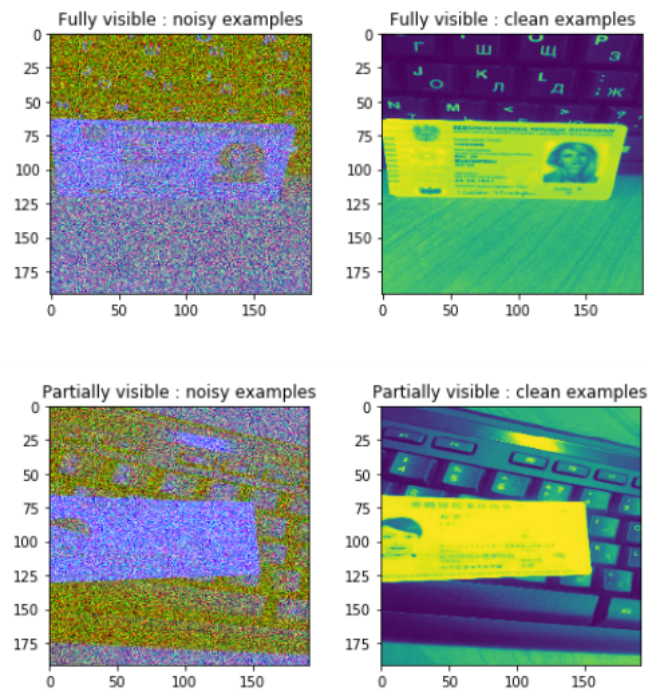
GICSD_1_1_33.png PARTIAL_VISIBILITY

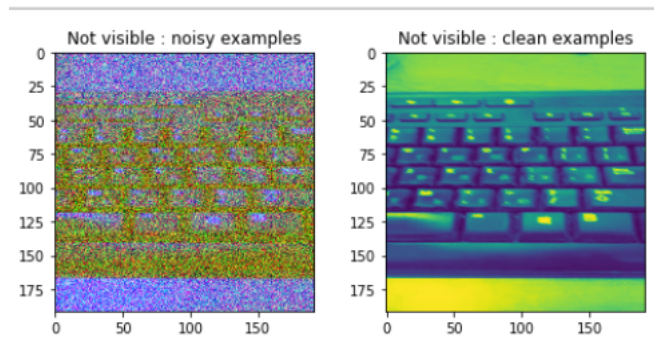
NO_VISIBILITY

GICSD_1_7_213.png Y

Some examples of the three classes are provided –

The clean image is provided in image[:, :, 2], i.e. the third channel, in case you want to use it as a ground truth for your solution.





The dataset is highly imbalanced 646 examples of fully visible examples, 123 partially visible and 31 not visible examples.

A supporting notebook titled 'Data_Exploration_Viz' is provided to visualize some data and parses the csv file and puts each examples into different folders based on the class labels, i.e. classes/FULL_VISIBILITY, classes/NO_VISIBILITY, classes/PARTIAL_VISIBILITY. This support csv file reader code and notebook is provided to you to speed up development and get you closer to the neural network faster. You may choose to ignore or inspire from the notebook and prepare your own parsing file, however we are not judging you based on your ETL skills.

Prepare a notebook for

- Deep Learning Classifier that inputs the noisy image and predicts the correct class label. The problem is quite complex therefore a classification accuracy atleast above 82% (i.e. 1.25% of the baseline performance of just choosing the dominant class) on the test data is desired.
- Properly comment your code
- Add visualizations and plots for performance metric that you have used as needed
- Provide rationales behind choosing a particular approach (loss function or architecture) to tackle a certain aspect in the problem
- Demonstrate your inference results on some test data, with ground truth and predictions of your model, along with confusion matrix or other metric you found more suitable for the problem.

Prepare the corresponding readme.md file for

- What the problem is about? Dataset used, and associated challenges.
- Describe your proposed solution.
- Future works and additional potential approaches to tackle the given problem.

Your submission should include

Solution notebook, and Readme.md, and any other associated files you used.

You are free to use any Python packages, use Keras, tensorflow or PyTorch. Don't worry about production-quality code, emphasis more on the model accuracy and the approach. More than data visualization and data exploration focus on the model, loss function and its' performance.

The assignment shouldn't take you more than **8 hrs** to complete.