Chapter 1

Interpretability

In this section, we will evaluate the model's prediction to better understand it. We will find if there are any features in the patches that can confuse it and if the model's output is robust. First, we will introduce on technique used to highlight the region of the input image that contribute the most to the model predictions. Then, we will use it on the data from the *Quarry* test set to find out the patches were the model fails and analyze them.

Later, we will work with custom created patches with different features, walls, bumps, etc, to test the robustness of the model by comparing its predictions to the real data gathered from the simulator.

1.1 Grad-CAM

Gradient-weighted Class Activation Mapping (Grad-CAM) [**gradcam**] is a technique to produce ẗisual explanationsf̈or convolutional neural networks. It highlights the regions of the input image that contribuite the most to the prediciton.

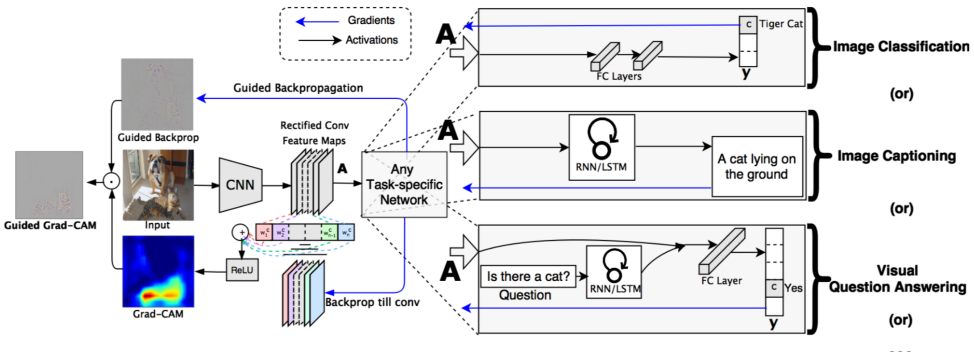

Figure 1.1. Grad-CAM procedure on an input image. Image from the original paper [**gradcam**].

In detail, the output with respect to a target class is backpropagate while storing the

gradient and the output in the last convolution. Then, global average is applyed to the saved gradient keeping the channel dimension in order to get a 1-d tensor, this will represent the importance of each channel in the target convolutional layer. We each element of the convolutional layer outputs is multiplyed with the averaged gradients to create the grad cam. This whole procedure is fast and it is architecture independent.

In detail, the output with respect to a target class is backpropagate while storing the gradient and the output in the last convolution. Then global average is applyed the saved gradient keeping the channel dimension in order to get a 1-d tensor, this will represent the importance of each channel in the target convolutional layer. We each element of the convolutional layer outputs is multiplyed with the averaged gradients to create the grad cam. This whole procedure is fast and it is architecture independent.

1.1.1 Features

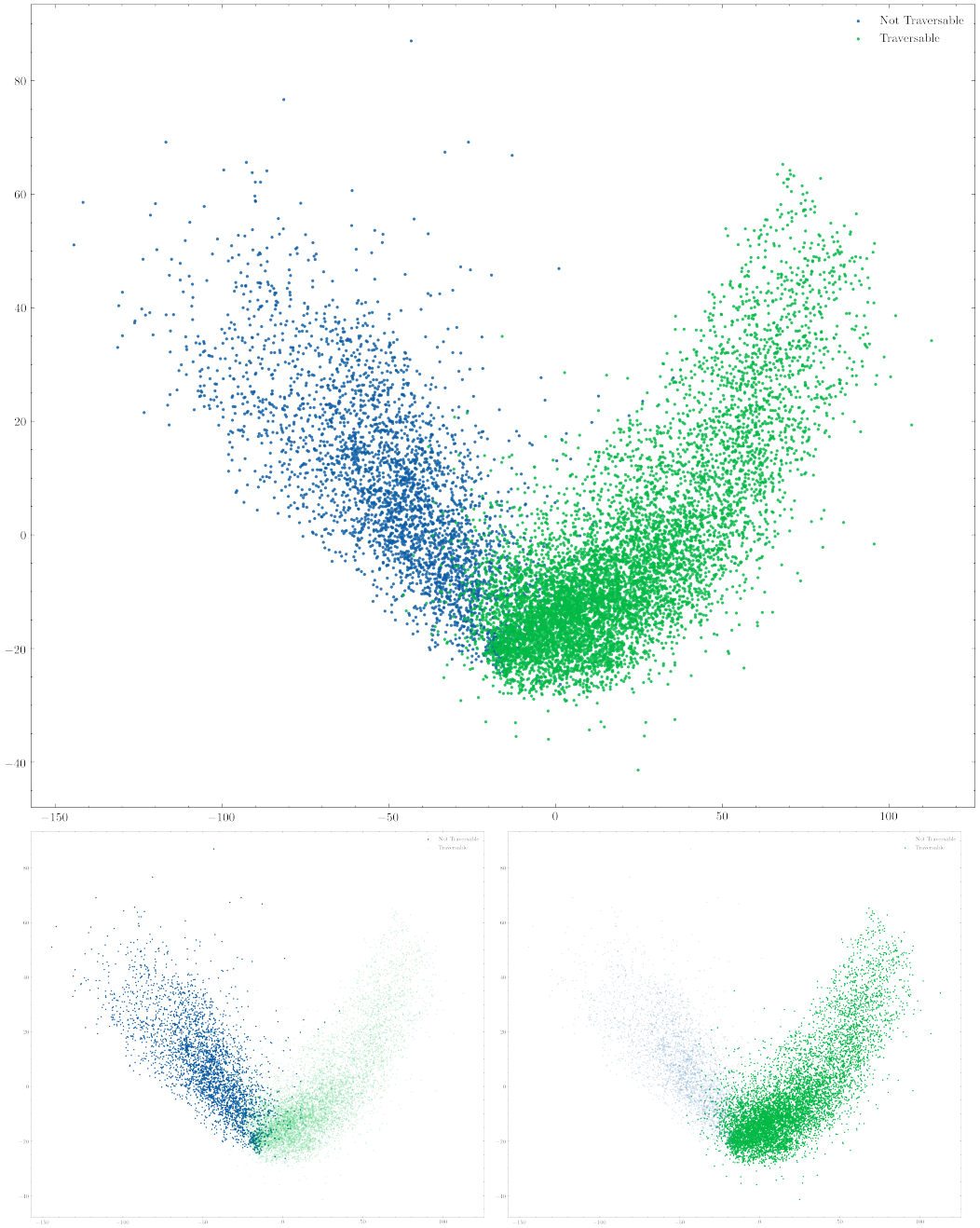

Figure 1.2. TODO

1.1.2 Features

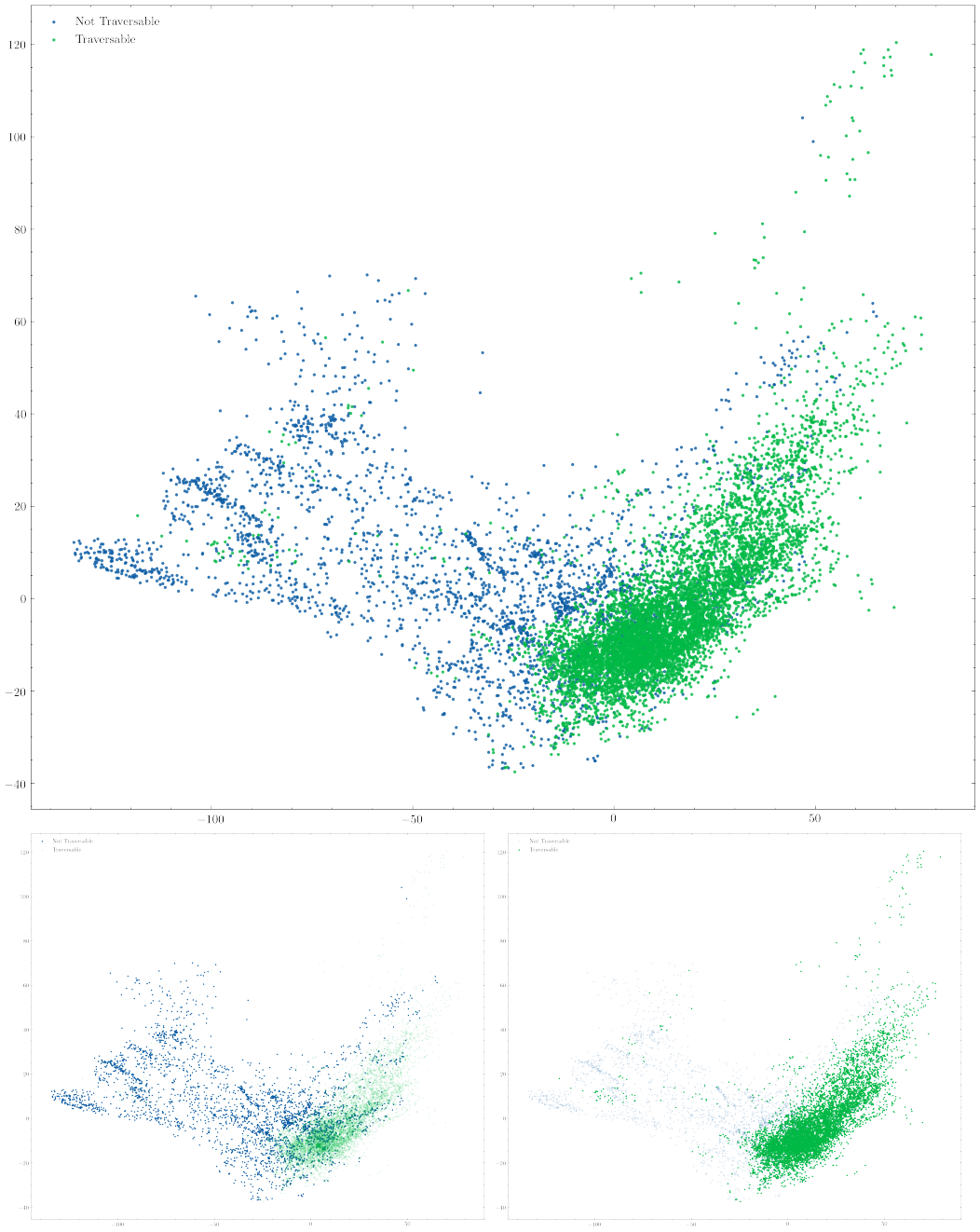

Figure 1.3. TODO