Team: Oliver Broadrick and Marshall Thompson

We will propose two projects since we don't have, based on our relative lack of experience with neural networks, a great sense for the feasibility of our primary idea. Therefore the second idea will serve as a backup in case the first one is deemed unreasonable by the instructor. We appreciate your feedback on both ideas, but we would like to do our first proposed project if it is reasonable for us to do so.

Project 1:

Implement, in TensorFlow, the keypoint recognition networks described in "R2D2: Repeatable and Reliable Detector and Descriptor" by Revaud et al. The described R2D2 points, computed for an input image, are found by neural networks trained to detect and describe the points while simultaneously producing a prediction of their descriminativeness (how likely they are to produce confident matches). Figure 1 (from the R2D2 paper) helps visualize the outputs of the R2D2 network.

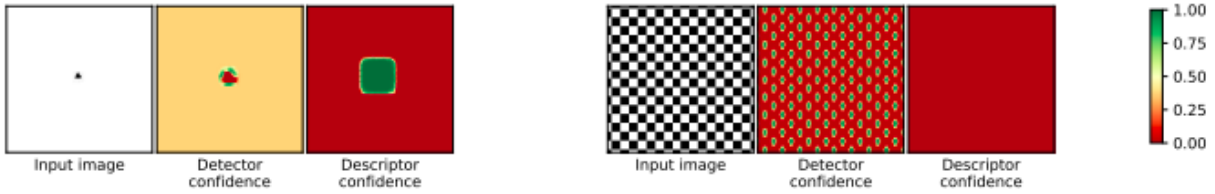

Figure 1. Toy examples to illustrate the key difference between repeatability (2nd column) and reliability (3rd column) for a given image. Repeatable regions in the first image are only located near the black triangle, however, all patches containing it are equally reliable. In contrast, all squares in the checkerboard pattern are salient hence repeatable, but none of them is discriminative due to self-similarity. Both confidence maps were estimated by our network.

Figure 2 (from the R2D2 paper) gives a diagram of the model implemented.

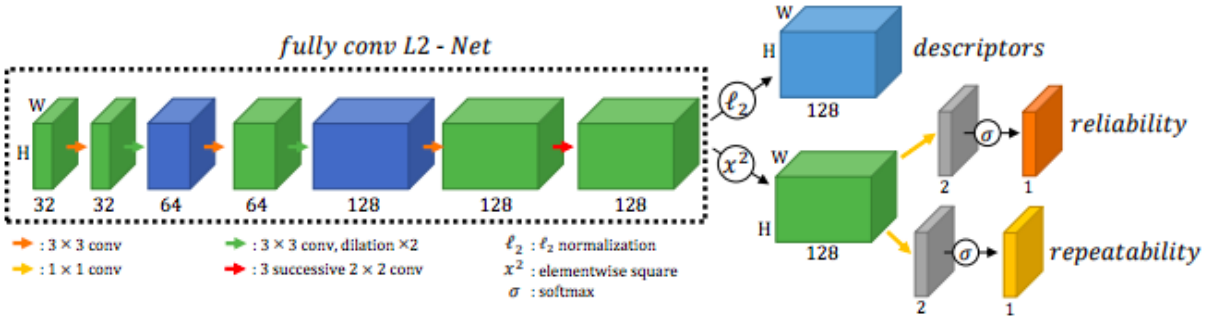

Figure 2. Overview of our network for jointly learning repeatable and reliable matches.

Our goal in this project would be to implement this model in TensorFlow. We see this as an adequate goal for the course project because the R2D2 network is a state-of-the-art computer vision neural network model. We look forward to hearing whether implementation should be feasible within the intended scope of our assignment.

Project 2:

If Project 1 is not feasible, we are interested in the different options for activation functions. For Project 2, we would like to implement a simple neural network model (perhaps training it on the original MNIST numbers of fashion dataset) and then compare the results of the model with various activation functions (relu, sigmoid, perhaps a function of our own, etc).