

In this Supplementary Material, we present the results of fine-tuning blocks 1 and 9 only in Fig. S1. We also employed Keras-Vis [1] to visualize the input image which maximize the activation in some example neurons. In Fig. S2, we show some examples features from a shallow (4th convolutional layer) and a deep layer (last convolutional layer) in the pre-trained network and the fine-tuned network when the whole network is fine-tuned. Note that these features are derived from the network on one fold of BUS dataset only. The features were found by a gradient search from a random image starting point. There are many features per layer, and it is unlikely that the local maximizer for a feature in one run (or fold) would be a local maximizer for any feature in another run (or fold). Therefore, it is not easy to interpret these features to comment on the impact of fine-tuning. It is however interesting to see low-level features in shallow layers and high-level features in deep layers.

[1] Kotikalapudi, R., contributors: keras-vis. <https://github.com/raghakot/keras-vis> (2017)

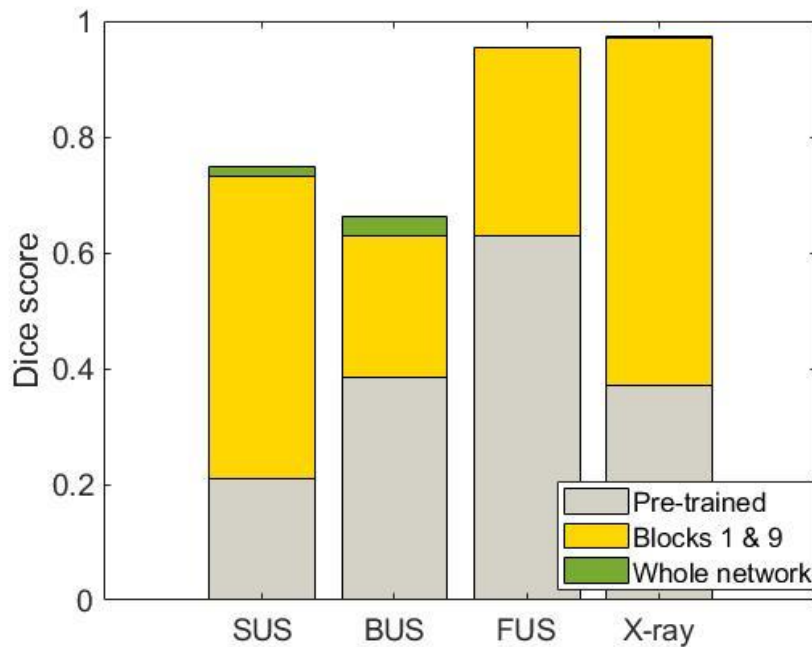


Figure S1. The impact of fine-tuning blocks 1 and 9, compared to fine-tuning the whole network. The gain added by fine-tuning the whole network (green portion) is not high, particularly, when the number of images in the training set is low. The figure depicts the example case of using 5% of the data for training.

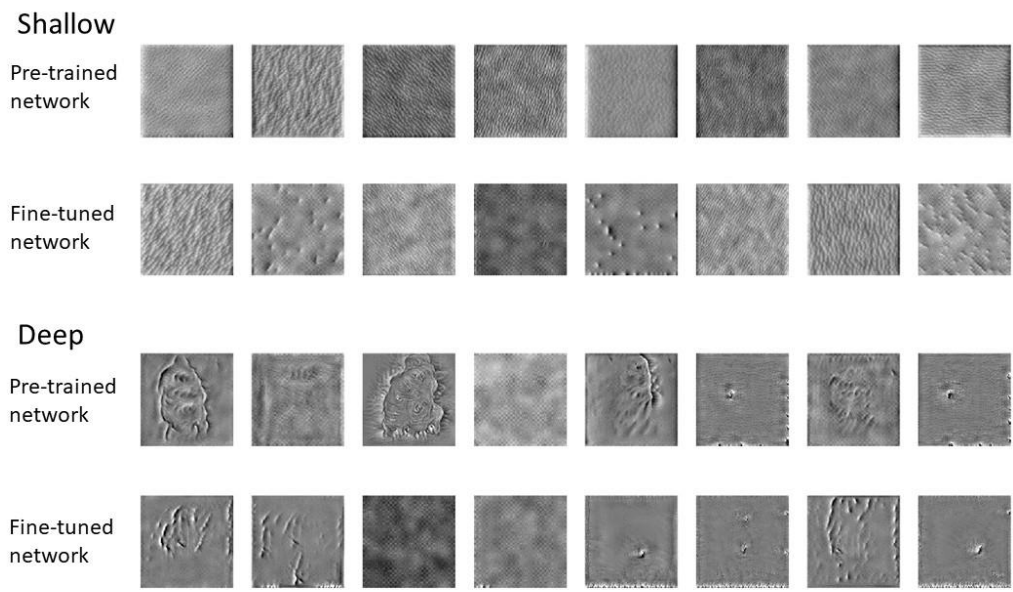


Figure S2- Feature visualization in a shallow and a deep layer. First row is showing the filters in the pre-trained network, and the second row is showing the same filters after fine-tuning the whole network on BUS dataset.