Adaptive Intermediate Representations for Video Understanding - Supplemental Materials

Juhana Kangaspunta, AJ Piergiovanni, Rico Jonschkowski, Michael Ryoo, Anelia Angelova
Google Research
{juhana,ajpiergi,rjon,mryoo,anelia}@google.com

1. Additional visualizations

In this section we show visualizations of the intermediate representations. Figure 1 shows the semantic segmentation intermediate representation both with and without gradients flowing for the Toyota Smarthome dataset.

2. Network architecture

In Figure 2 we visualize the network architecture which is simple, yet efficient.

3. Evolutionary search progression

In Figure 3 we visualize the progression of the evolutionary search for the optimal loss weights. The graph shows the accuracy of the best found model as a function of total iterations, i.e. models evaluated. The evolution was done on the HMDB-51 dataset (without pretraining). As seen, not too many iterations are needed to converge to a good solution.

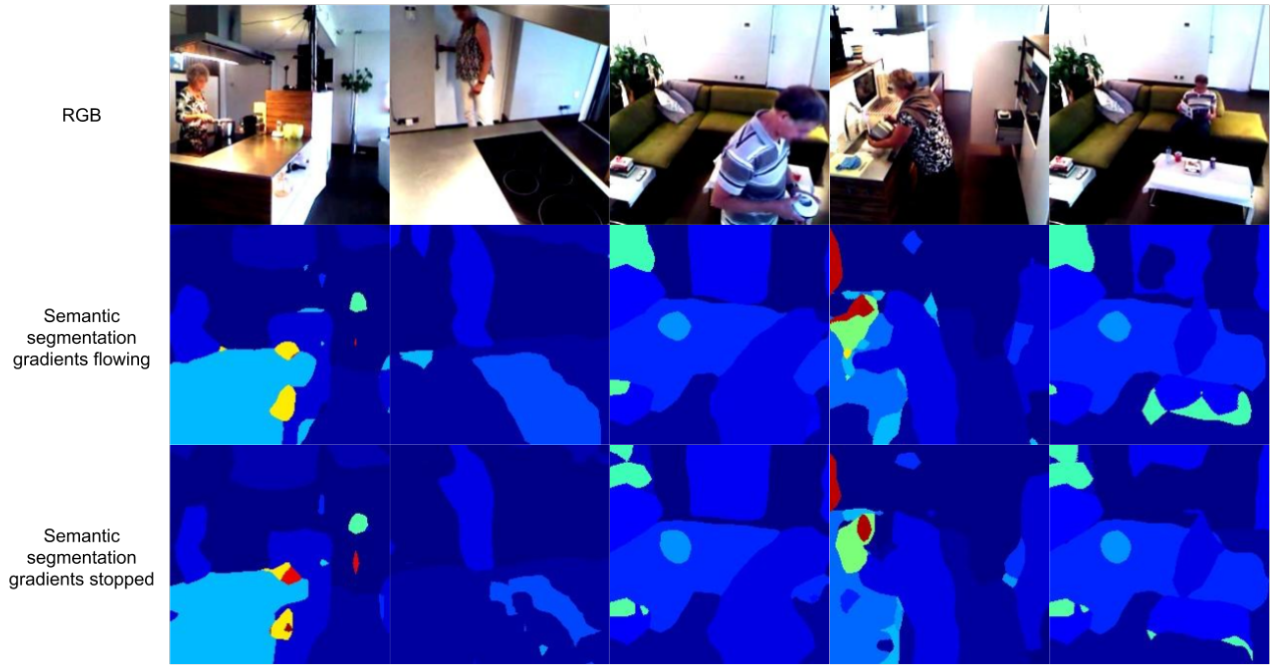

Figure 1. Predictions of the semantic segmentation mini-network. First row shows an RGB input frame. Second row shows a semantic segmentation prediction from a network where gradients were flowing from the video classification losses all the way through the semantic segmentation task. Third row shows a semantic segmentation prediction from a network where video classification gradients were stopped at the semantic segmentation logits.

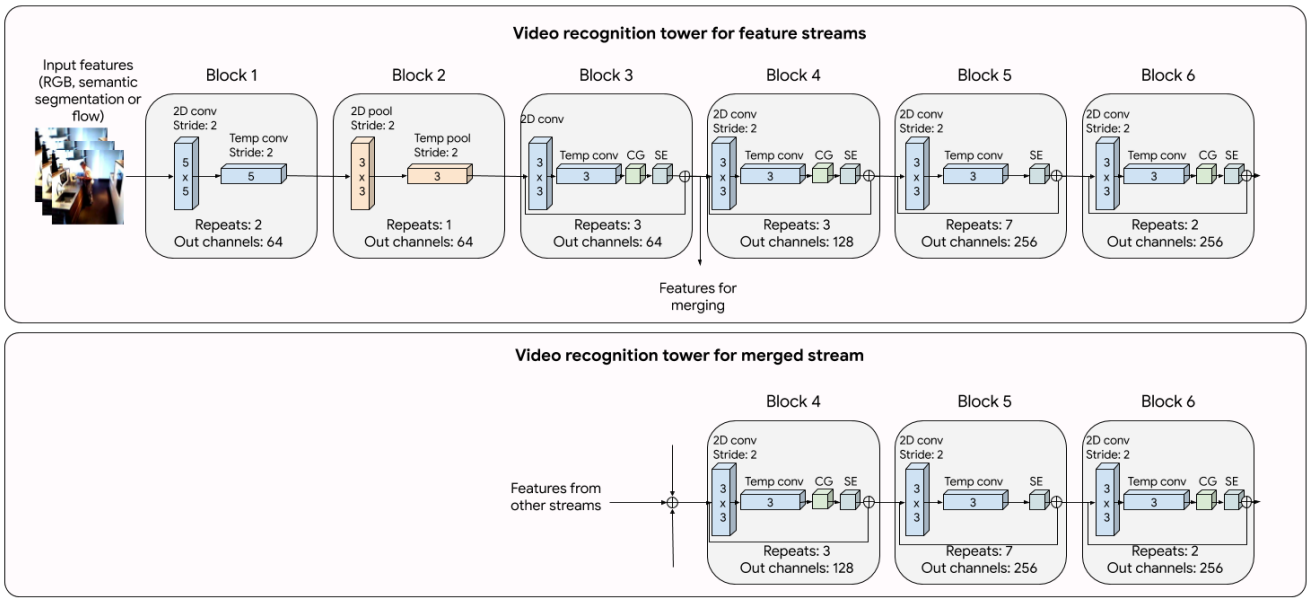

Figure 2. AIRStreams video recognition tower architecture. Each individual feature tower (RGB, flow and semantic segmentation) is an instance of the network displayed in the top section. The tower for the merged stream is displayed in the bottom section and takes as input the features from Block 3 of all the feature streams. The merged stream tower is identical to the Blocks 4-6 of the feature streams. The network is built from six blocks, each repeated one or more times. Each block is assembled from the following layers: 2D convolution over the spatial dimensions (2D conv), 1D convolution over the temporal dimension (Temp conv), max-pooling over either the spatial or temporal dimensions (2D pool, Temp pool), context-gating layer (CG), and squeeze-excite (SE) layer. The kernel size of each layer is written inside the illustrating box. Skip connections over each block are illustrated with connecting lines.

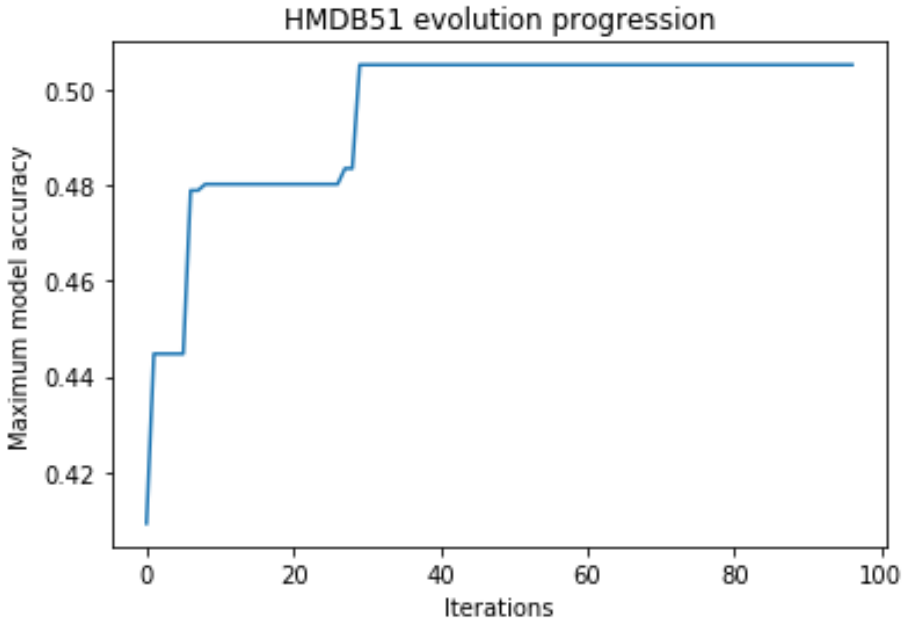

Figure 3. Accuracy of the best found model as a function of the number of iterations.