Supplementary Material: Hide-and-Seek: Forcing a Network to be Meticulous for Weakly-supervised Object and Action Localization

Krishna Kumar Singh and Yong Jae Lee
University of California, Davis

In this document, we provide: 1) discussion on the failure cases of HaS for object localization, 2) image classification results of HaS on challenging images, and 3) qualitative action localization results. We next discuss each in turn.

1. Failure cases of Hide-and-Seek

Both the quantitative (Table 2) and qualitative results (Figure 4) in the paper show that overall our Hide-and-seek (HaS) approach leads to better localization compared to the GAP baseline. Still, HaS is not perfect and there are some specific scenarios where it fails and produces inferior localization compared to GAP.

Figure 1 shows example failure cases of AlexNet-HaS compared to AlexNet-GAP. In the first two rows, HaS fails to localize a single object instance because there are multiple instances of the same object that are spatially close to each other. This leads to our approach merging the localizations of the two object instances together. For example in the first row, our localization of the two lab coats are merged together to produce a bigger bounding box containing both of them. In contrast, AlexNet-GAP produces a more selective localization (focusing mainly on only the lab coat on the right), which leads to a bounding box that covers only a single lab coat. In the third and fourth rows, failure occurs due to the strong co-occurrence of the contextual objects near the object-of-interest. Specifically, in the third row, our AlexNet-HaS localizes parts of the house (context) along with the fence (object-of-interest) because house co-occurs with fences frequently. As a result, when parts of the fence are hidden during training the network starts to focus on the house regions in order to do well for the fence classification task. Finally, in the last row, our AlexNet-HaS localizes both the bird and its reflection in the water, which leads to an incorrect bounding box.

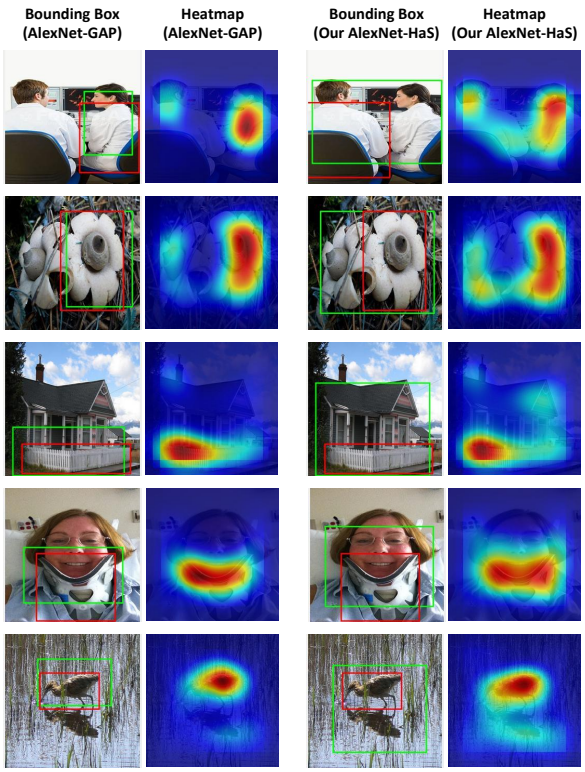

Figure 1. Example failure cases of AlexNet-HaS. For each image we show the bounding box (red: ground-truth, green: predicted) and CAM obtained by the AlexNet-GAP baseline (left) and our AlexNet-HaS approach (right). In the first two rows, our method fails due to merging of the localization of multiple instances of the object-of-interest. In the third and fourth rows, it fails due to strong co-occurrence of contextual objects with the object-of-interest. In the last row our localizer gets confused due to the reflection of the bird.

2. Classification of challenging images

In our Hide-and-Seek (HaS) approach, the network is trained using images in which patches are hidden randomly. This gives the network the ability to classify images correctly even when the objects are partially-occluded and when its most discriminative parts are not visible. In Figure 2, we show challenging cases for which AlexNet-GAP fails but our AlexNet-HaS successfully classifies the images. Our AlexNet-HaS can correctly classify 'African Crocodile' and 'Notebook' by just looking at the leg and keypad, respectively. It can also classify 'German Shepherd', 'Ostrich', 'Indri' and 'Rottweiler' correctly without

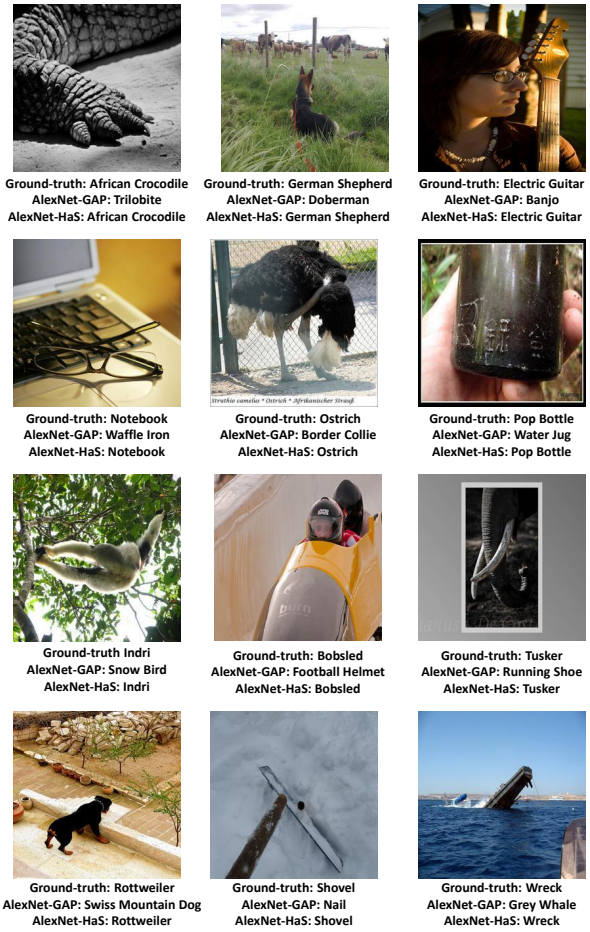

Figure 2. Comparison of our AlexNet-HaS vs. the AlexNet-GAP baseline for classification of challenging images. For each image, we show the ground-truth label followed by top class predicted by AlexNet-GAP and AlexNet-HaS. AlexNet-HaS is able to classify the images correctly even when they are partially occluded (African crocodile, electric guitar, notebook, pop bottle, bobsled, tusker, shovel and wreck). Even when the most discriminative part is hidden, our AlexNet-HaS classifies the image correctly; for example, the faces of the German Shepherd, ostrich, indri and rottweiler are hidden but our AlexNet-HaS is still able to classify them correctly.

looking at the face, which is the most discriminative part. Quantitatively, on a set of 200 images (from 100 random classes) with partial-occlusions, our AlexNet-HaS model produces 3% higher classification accuracy than AlexNet-GAP.

3. Action localization qualitative results

Finally, in Figure 3, we compare the temporal action localization results of our approach of randomly hiding frame segments while learning an action classifier (Video-HaS) versus the baseline approach of showing the whole video during training (Video-full). For each action, we uniformly sample the frames and show: 1) Ground-truth (first row, frames belonging to action have green boundary), 2) Video-full (second row, localized frames have red boundary) and 3) Video-HaS (third row, localized frames have red boundary).

From Figure 3, we can see that our Video-HaS localizes most of the temporal extent of an action while Video-full only localizes some key moments. For example, in the case of javelin throw (second example), Video-HaS localizes all the frames associated with the action where as Video-full only localizes a frame in which the javelin is thrown. In the third example, Video-full localizes only the beginning part of high jump while Video-HaS localizes all relevant frames. In last row, we show a failure case of Video-HaS in which it incorrectly localizes beyond the temporal extent of diving. Since frames containing a swimming pool follow the diving action frequently, when the diving frames are hidden the network starts focusing on the context frames containing swimming pool to classify the action as diving.

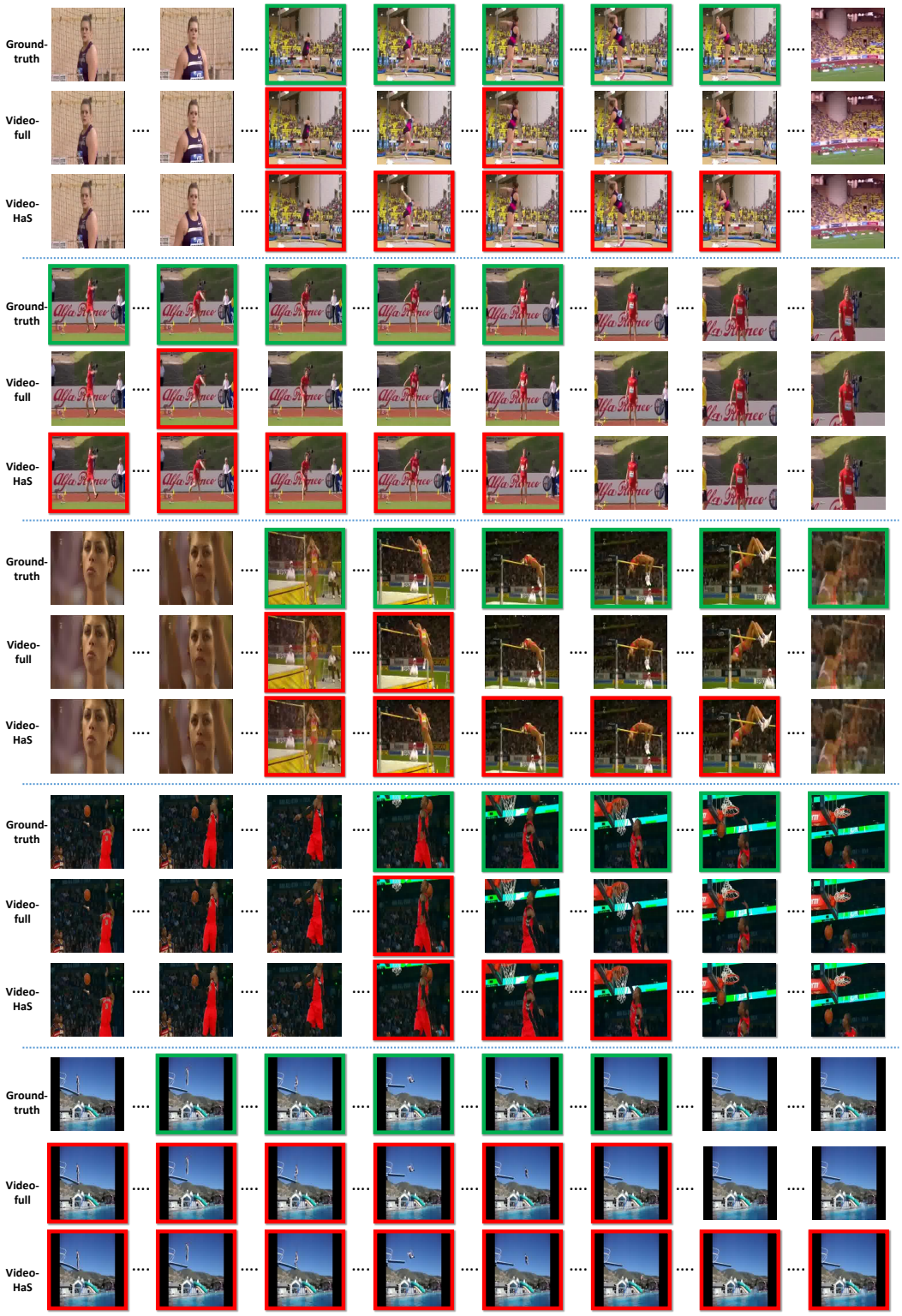

Figure 3. Comparison of action localization between the Video-full baseline and our method of Video-HaS. For each action, we uniformly sample the frames and show the ground-truth in the first row (frames with a green boundary belong to the action), followed by the Video-full and Video-HaS localizations (frames with a red boundary). For each action (except the last one), Video-HaS localizes the full extent of the action more accurately compared to Video-full, which tends to localize only some key frames. For example in the third example, Video-full only localizes the initial part of high-jump whereas Video-HaS localizes all relevant frames. In the last example, we show a failure case of our Video-HaS, in which it incorrectly localizes the last two frames as diving due to the co-occurring swimming pool context.