LaTeX Weakly Supervised Point-to-tell Neural Network

Fred Lu
New York University
New York, NY, USA
dl3957@nyu.edu

Abstract

[3] introduced an assistive system for the visually impaired that, using images from a head-mounted camera, informs the user what object is being pointed at by the user's hand, and its relative location to the hand. In [3], the authors used a fully supervised, Single Shot Detector-based object detection network to calculate bounding boxes for objects of interest and the user's hand. However, creating the ground truth for the model can be costly and time-consuming, as it would require human annotators to draw precise bounding boxes for each ground-truth image. This paper proposes an alternative, weakly supervised model that accomplishes the same main function of the of point-to-tell assistive technology as presented in [3] (without recognizing the distance of objects from the user), by adding Grad-Cams [1] and an Attention Mining Loss [2] to an image classification network. Specifically, the annotation only involves asking what objects are present in a given image, which can be done efficiently by either humans or a different neural network capable of classifying images with high accuracy.

1. Introduction

1.1. Background

[1] proposed a way to calculate to provide visual explanations for a convolutional neural network based on gradients, named Grad-Cam. The Grad-Cam has the same width and height as the input image, but has only one channel, which indicates how important a pixel is for the neural network's output. Hence, we can train an image classification neural network to identify if the user's hand and objects are present in an image, and then use Grad-Cam to localize the hand and each object that is present. Using the localization cues, we can then inform the user what he or she is pointing at, and where the object is relative to the user's hand. Thus, we can implement point-to-tell with weak supervision, i.e. without the need for pixel-level segmentation images as ground-truth.

1.2. Issues with object/hand localization using Grad-Cam

Grad-Cam is a method for visualizing how important each part of the input image is to the output value by back-propagating gradients. In my case, I set up a Grad-Cam for my classification network so that it outputs an image that has the same dimensions as the input image but has only one channel, which has a value between 0 and 1 that indicates how important each pixel in the original image to producing the output confidence of a specific class. Although Grad-Cam can be interpreted as segmenting an image by object categories, it is often includes false or biased features of objects in addition to true features. For example, (Figures to be inserted – real-world test data showing the false features included by Grad-Cam). As a result, the "segmentation" produced by Grad-Cam is noisy and cannot localize objects with high accuracy.

1.3. Self-supervised Grad-Cam Optimization

For the aforementioned issues, [1] proposed a way of making Grad-Cams focus more on true features of an object without needing to modify the architecture of the existing neural network. By introducing the Attention Mining loss, we can guide the image classification network to produce Grad-Cams that focus more on true features, which also enhances generalization performance of image classification.

1.4. Our Method

By combining an image classification neural network with Grad-Cams and the Attention Mining loss, we end up with a model that is able to localize objects with high accuracy under weak supervision. Such a network is also easy to train and implement because it does not need extra parameters for a true image segmentation function.

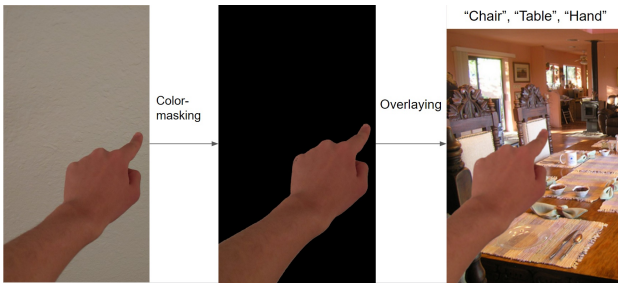


Figure 1. Method in which the hand overlay was created

2. Implementation Details

3. Architecture

The design is very similar to what the original GAIN paper proposed [2], with a modified attention mining loss to speed up training. Specifically, the attention mining loss was modified from being the average of the classification confidence of the masked image for every classes to the classification confidence of the masked image of a single class randomly chosen from the ones that were present in the ground truth image. The backbone used is FCN-8s [5]. I used a version of FCN-8s that was pre-trained on the VOC SBD Dataset, which was avaiable on this Github Repository. In order to allow FCN-8s to output classification confidences, I applied 2x2 max-pooling and global average pooling to the output of FCN-8s's last convolutional layer, which resulted in a feature map of size 1x1 and 512 channels. I then attached three fully connected layers to the resulting feature map. The fully connected layers have shape [512 (number of channels of the 1x1 feature map)→4096], [4096→4096], [4096→21 (number of image classes excluding background)]. The layers that were dedicated to semantic segmentation from the FCN-8s network were not used at any stage of training or testing.

3.1. Dataset Used

The point-to-tell model was trained using only image-level labels (which objects are present in an image) on a subset of the VOC SBD Dataset. Segmentation images were converted to image-level labels by calculating the unique pixel-wise classes that appeared in a segmentation image. The training subset were selected as the images of the VOC SBD Dataset that included at least one of the following object categories: bottle, chair, table, person, sofa, tv/monitor. Then, for each of the image, I made an extra copy of the same image with a superimposed patch of a hand pointing at one of the randomly chosen objects that is present in the original VOC image. The hand-overlay process is shown in figure 1 The entire synthetic dataset contained 12656 images.

3.2. Training

Two data augmentation techniques were used across the whole training process: each training image had a 50% chance of being horizontally flipped and all images had PCA noisy added in the same fashion as used in training AlexNet [4]. The training is done in two stages. In the first stage, the network was trained with all layers frozen except the 3 fully connected layers used for classification, and only the classification loss (Sigmoid cross entropy) was used. A dropout ratio of 0.5 was also applied to the first two of the three FC layers to prevent over-fitting. I used the ADAM optimizer with an initial learning rate of $5 * 10^{-}5$, and the training lasted for 47 epochs on the VOC+Hand dataset. The loss converged relatively quickly (considering we're only training a 3-layer classification network based on existing features maps from the CNN. loss, the loss converged relatively quickly. The purpose of this stage of training was to acquire a baseline network that was capable of classifying images but whose activation patterns had not been tuned with the attention mining so we can use it for later comparisons. Additionally, training using both the classification and attention mining loss from scratch could be too difficult to learn and result in slow or unstable training (though I did not thoroughly test this hypothesis).

In the second stage, I started with the model given by the the first stage of training, and trained it with both the classification loss and the modified version of the attention mining loss mentioned in the beginning of this section. Stage two lasted for 20000 iteration (slightly less than 2 epochs as the dataset consisted of 12656 images). I used the ADAM optimizer with an initial learning rate of $5 * 10^{-}7$. After stage two finished, improvements were seen in the accuracy of the attention maps, which means the network has learned to focus more heavily on "true" or "distinctive" features and should thus result in better generalizability, as the GAIN paper has theorized [2]. However, stage two training also resulted a very minor increase in the classification loss.

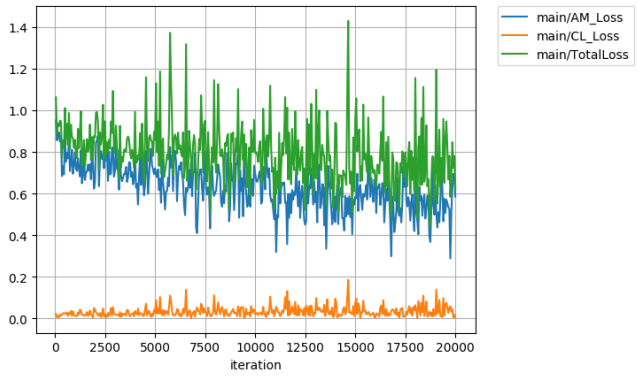


Figure 2. An input image and the output attention maps of two versions of the point-to-tell model.

3.2.1 Implementing Point-to-tell using Network's Attention Maps

After acquiring a network that had reasonable attention map outputs, we could treat the attention map as a naive version of a semantic segmentation after applying a threshold mask. Shown in figure 2.

Then, using openCV, we can calculate the outer contours around each attention cluster. At this point, we take the topmost point in the contour for "hand" as the location of the finger. For other object categories, the contours were reduced to rectangular bounding boxes. In the final stage, the system determined the distance (magnitude and direction) between the finger and each object present in the image using the distance from the finger to the closest edge of the object's bounding box. The bounding boxes are somewhat noisy with the current model design and training methodology.

4. Experimentation Results

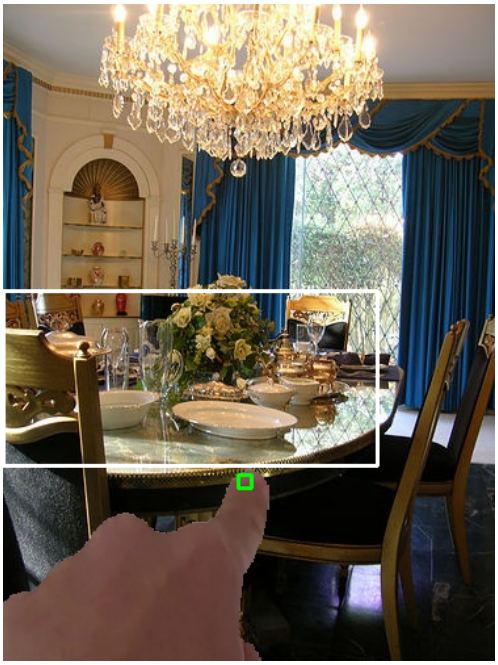
4.1. VOC SBD Validation Split + Different Hands: Image Classification

The model classifies every object category using a confidence threshold of 0.5. There are 2000 validation images. Similar to the training split, the validation images were generated from select of the VOC SBD dataset. Like the training split, half of the validation images are copies of the original half but with a superimposed patch of a hand. Below is the table showing the true positive, false negative, true negative, and false positive numbers, out of 2000 images.

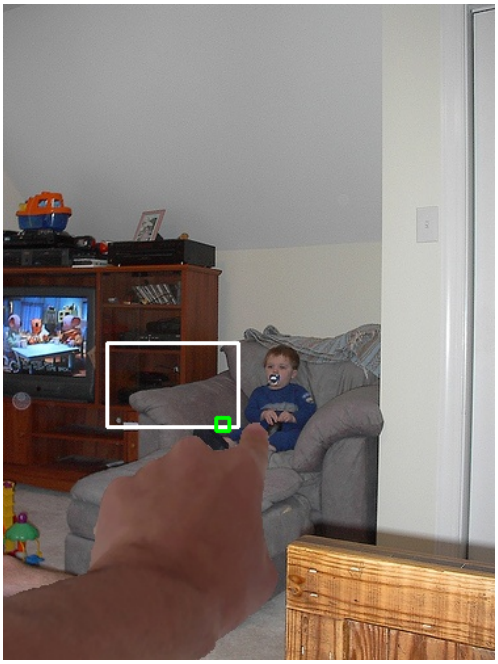
Obj. Class	Bottle	Chair	Table	Person	Sofa	TV
TP	86	387	320	1384	290	110
FN	332	279	208	86	80	158
TN	1565	1267	1419	519	1545	1711
FP	17	67	53	11	85	21

4.2. VOC SBD Validation Split + Different Hands: Bounding Box Precision

Below I show the images where hands are detected. In each image, the location of the finger is designated by a small green rectangle and the closest object is the designated by a white rectangle. The bounding boxes are generally accurate, however, sometimes the actual closest object to the hand is not being detected by the classifier so its bounding box is ignored. In all except the last image, the object detected class was correct (in the last image, the program mistook the dining table for chair).

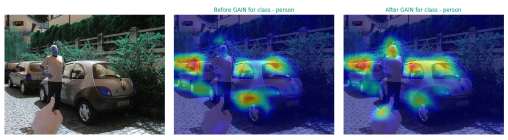


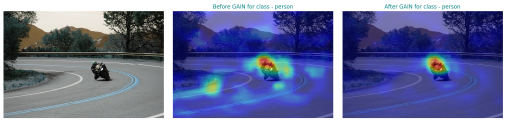
In this additional image below, the finger location was off so the sofa was marked as the closest object instead of the person.



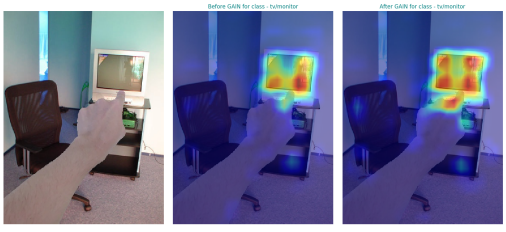
4.3. VOC SBD Validation Split + Different Hands: Attention Visualization

The following are the visualization of the Grad-Cams which are used to calculated object locations. Each image has three columns: the original image, the attention map (Grad-Cam) generated by the model trained without using the Attention Mining Loss, and the attention map generated by the model trained with the attention mining loss. The title for each column indicates the class with the highest output confidence.





These two images show that training with attention mining loss improves the concentration and correctness of attention maps.



The last four images are interesting because it shows that the network learns to consider the hand a part of the object only when the hand is pointing (overlapping) with the object. The same pattern is observed across other object categories such as chairs and people as well.

References

[1] Grad-cam:visual explanations from deep networks via gradient-based localization.

[2] Tell me where to look: Guided attention inference network.

[3] Wenjun Gui, Bingyu Li, Shuaihang Yuan, et al. An assistive low-vision platform that augments spatial cognition through proprioceptive guidance: Point-to-tell-and-touch.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[5] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016.