Improved Weakly-Supervised Object Localization Using Hide-and-Seek Based Deep Neural Network

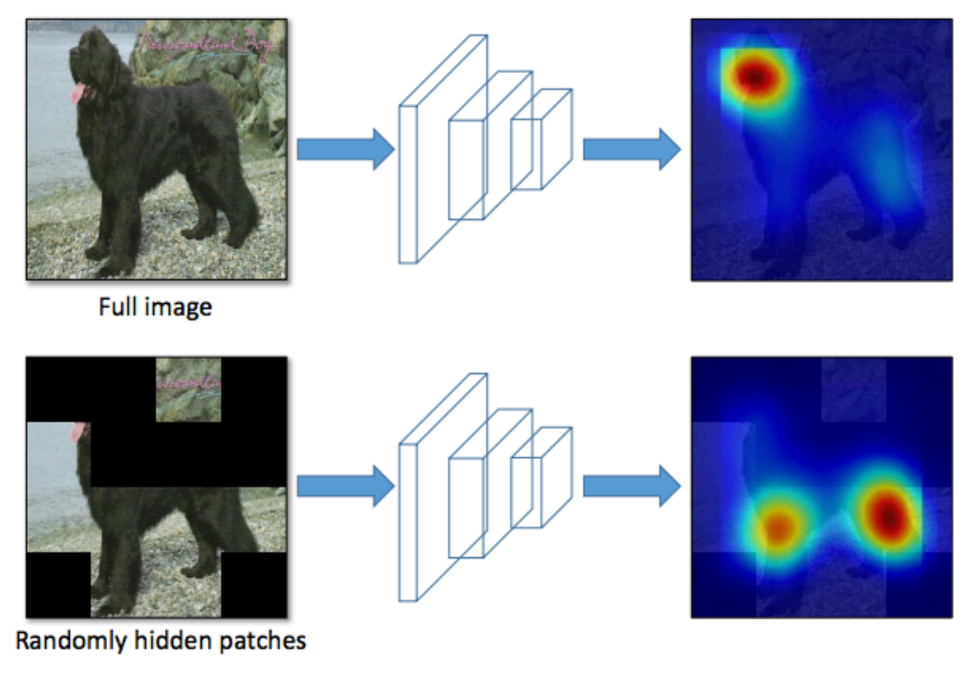
Gautam Pradeep
Mentor: Krishna Kumar Singh
Faculty Advisor: Prof. Yong Jae Lee
EnvironMentors-**AggieMentors**
University of California Davis / Mira Loma High School

Abstract

The goal of this project is to create a visualizing tool for a deep neural network that enhances object localization in images through a weakly-supervised framework. We will be using the Hide-and-Seek (HaS) approach to train a classifier which focuses on all the relevant regions of a class/object instead of solely the most discriminative parts. By randomly hiding patches of the image, the network is forced to see other less relevant regions, which allows for greater performance during testing. We obtain a Class Activation Map (CAM) in order to verify what regions of the image the detector is focusing on to most effectively classify and localize the object-of-interest. The visualizing tool created will give the opportunity for a user to upload any image and see the detection results and class activation maps for various classes. The user will train the hide-and-seek model on their custom training data to see real-time object localization. We create a PyTorch program in which users can train their own HaS network with their own dataset using this soon-to-be publicly available program.

Background Information

- **Weakly-supervised** models are actively studied for object localization/detection tasks.
- Weak supervision does not require bounding box annotations for training and relies solely on class labels which are cheaper to obtain, so larger datasets can be used
- Most existing models localize only the most discriminative parts of an object rather than all the relevant regions which results in misclassification and incorrect localizations
- The state-of-the-art ***Hide-and-Seek* model (HaS)** attempts to achieve superior performance by randomly hiding patches in a training image, forcing the network to see other relevant regions if and when the most discriminative part is hidden

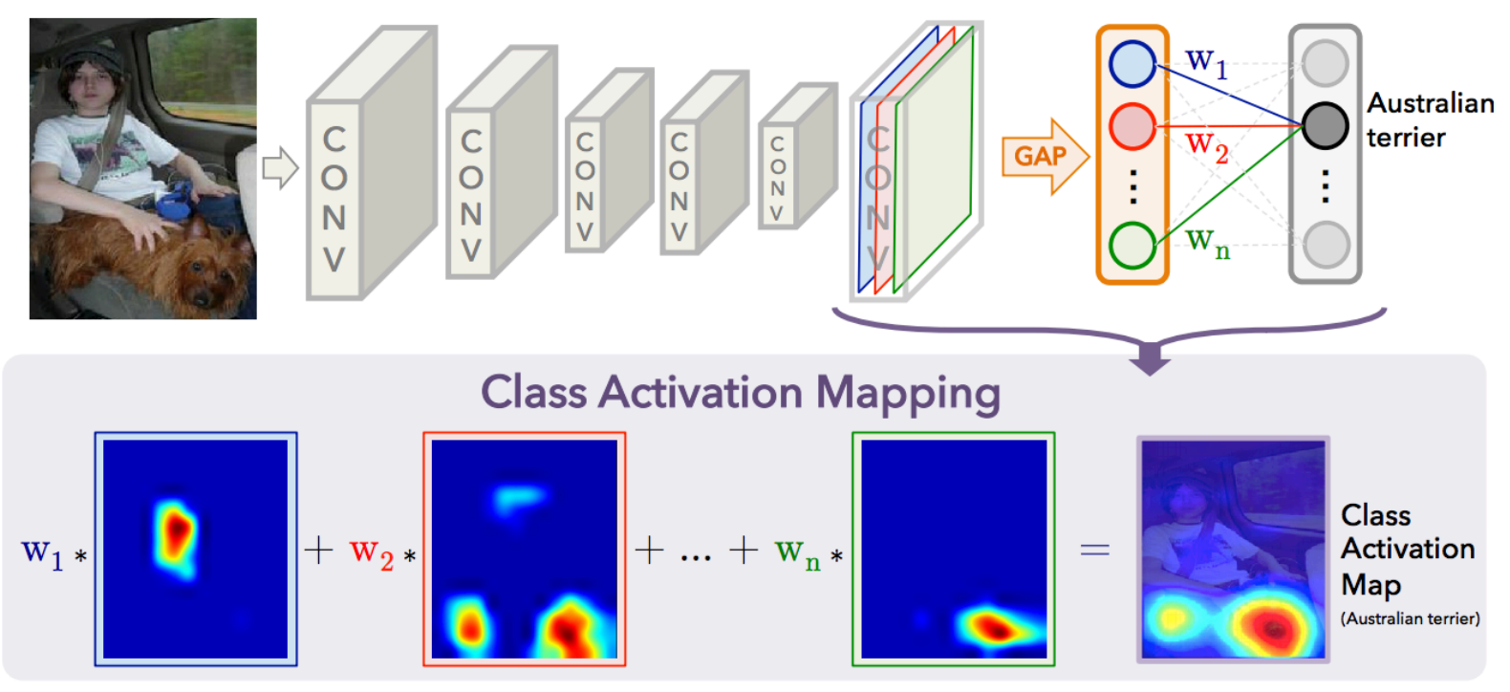


[1]

- The image is divided into a grid of equal-sized patches→ each patch is randomly hidden with some probability and given as input to a convolutional neural network (CNN) to learn image classification
- Every epoch of training, the hidden patches change randomly
- During testing, the full image (no hidden patches) is given as input to CNN

Class Activation Maps

- A **Class Activation Map (CAM)** for a particular category indicates the discriminative image regions used by the CNN to identify that class
- Through Global Average Pooling on the convolutional feature maps and the use of those features for a final fully-connected layer, we can find the **weights** of the output layer and determine what regions of the image were more discriminative in the classification and localization.



$$CAM(c, I) = \sum_{i=1}^{M} W(c,i) * F_i(I)$$ [2]

where *F* denotes the feature map for the *i*th layer of the CNN which has a weight *W*, providing the *CAM* for the class *c* in the image *I* [2]

- CAMs can give us "behind-the-scene" views into what the network is looking at when classifying the image, and we can then start to understand how the network independently is able to learn
- Hide-and-Seek model can be altered to accommodate for any dataset and network, then any user can implement the model with their own desired data to train a network for classification and localization, while also creating the CAMs to better understand the inner-workings of the neural network.

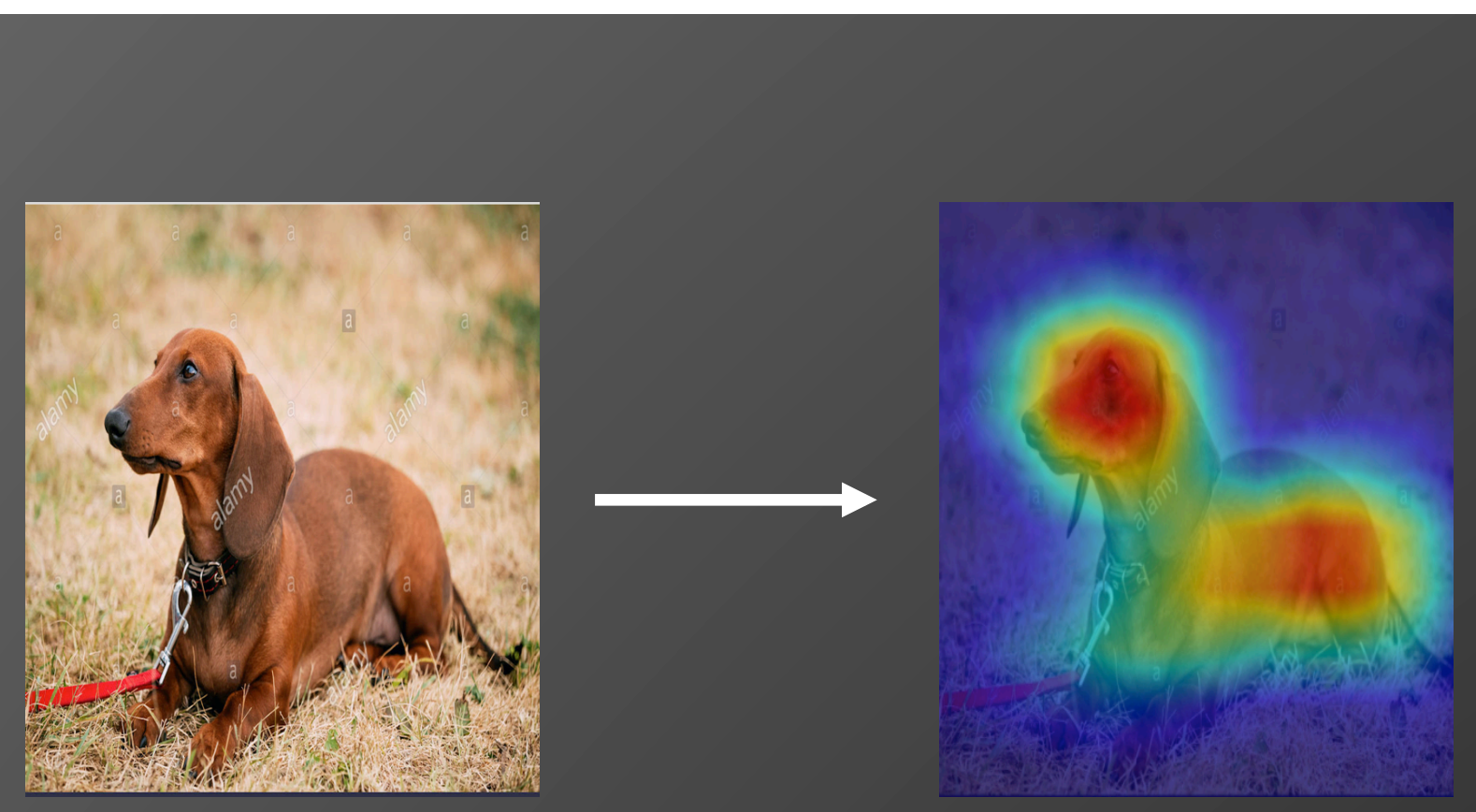
Question/Objective

The goal is to develop a visualization tool to users to remotely upload images and receive class activation maps for the corresponding images from the results of a deep neural network. The users should be able to choose what deep neural network to use and what class (category) they desire the CAM to represent.

The next objective is to create a program in a Python-based deep learning platform known as PyTorch to code for the Hide-and-Seek model and allow users to submit their data as input for the network to train. The goal was to make the model versatile so any user could use any data and have a working model to classify and localize objects of class.

Hypothesis

By visualizing the the object classification through class activation maps, we can qualitatively and quantitatively verify that the regional patch-hiding approach in *Hide-and-Seek* performs superior to the standard networks such as AlexNet and GoogLeNet



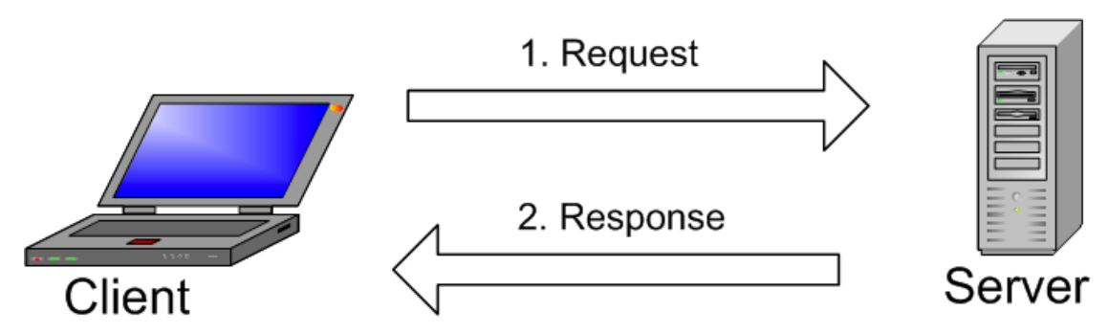
Methodology and Stages of Experimentation

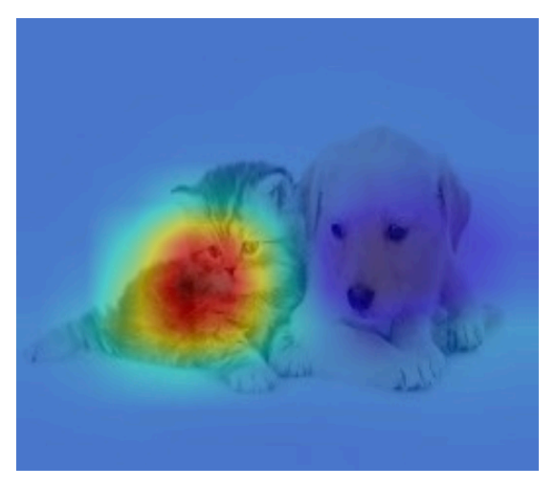
GENERAL

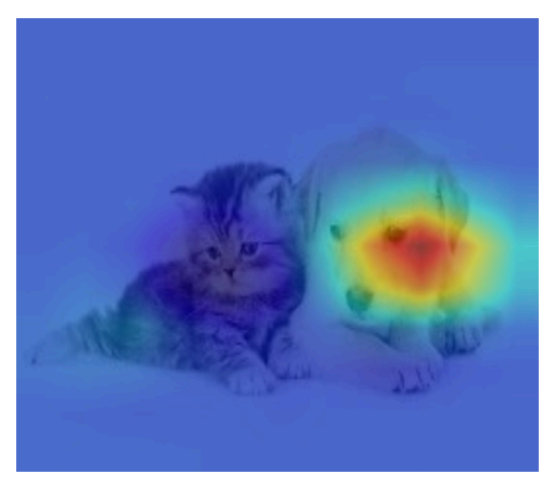
PHASE 1 : Class Activation Map Generation

MAJOR STAGES

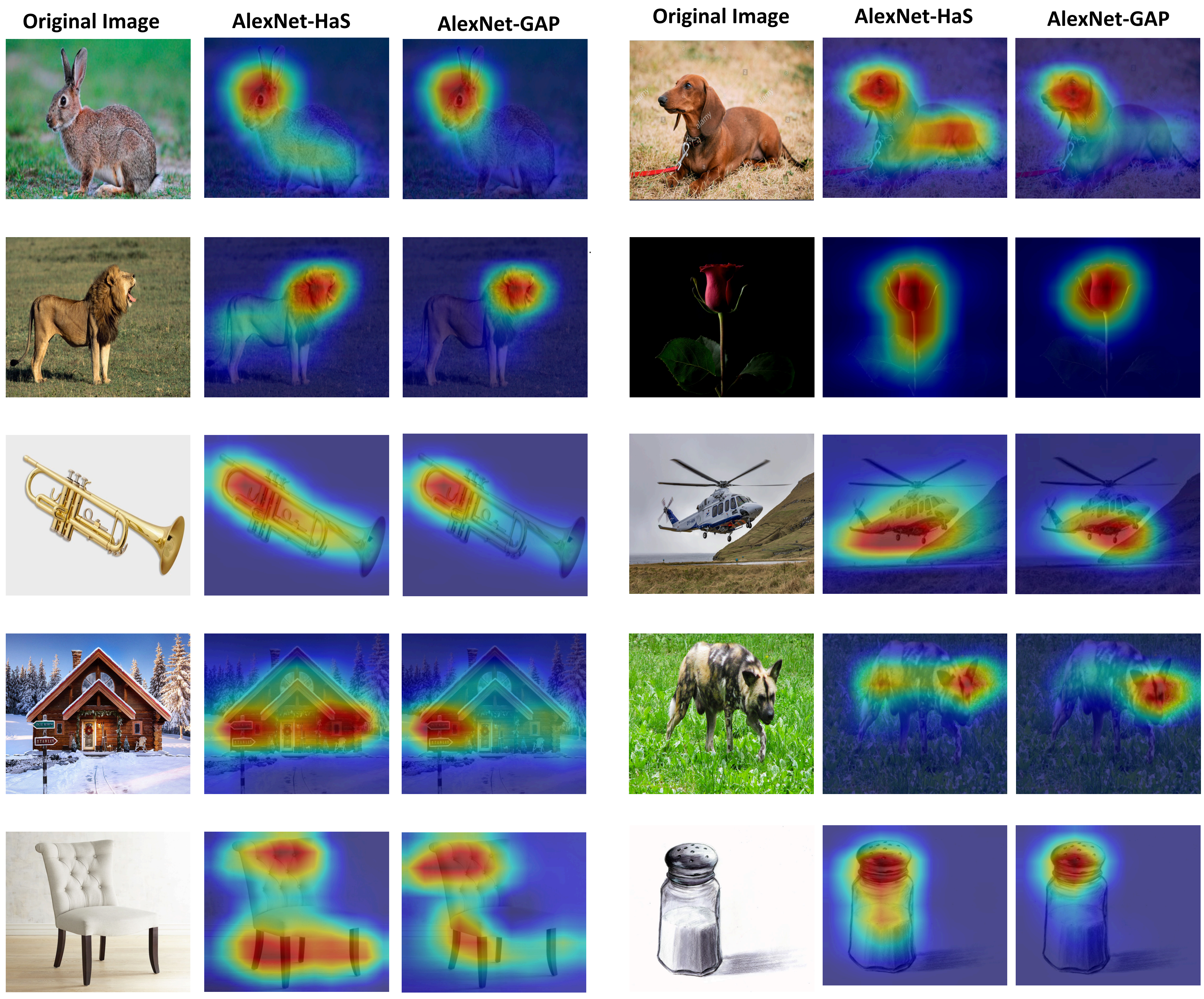
1. Develop socket connections between client and server sides to send files via a specific port channel
2. Create a web application to upload an image to send to server for further tasks
3. Utilize Hide-and-Seek model on the server to localize and classify the object-of-interest in an inputted image
   i) Allow option for user to run Hide-and-Seek network or AlexNet network
   ii) Allow user to choose a specific class for which to generate a CAM
   iii) Allow application to display top predictions (top 5) along with the CAM responding to the image
4. Generate CAM image and send CAM along with classification results back to client through HTML browser to display a list of top predictions




CAM localizing cat class


CAM localizing dog class

PHASE 1 QUALITATIVE RESULTS

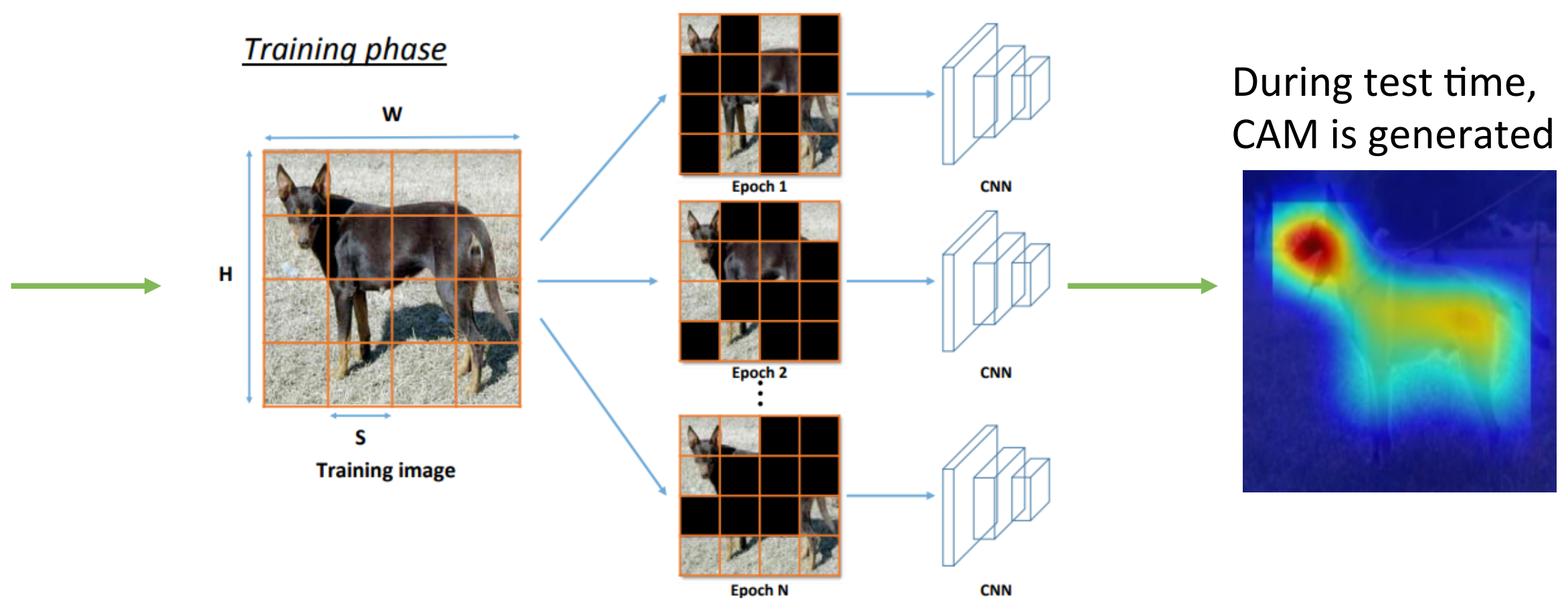


PHASE 2 : Developing PyTorch Program for HaS

MAJOR STAGES

INPUTS

1. **Training Images**
2. **Labels**
3. **Parameters: model, grid size, hiding probability**



During test time, CAM is generated

References

[1] Singh, Krishna Kumar, and Yong Jae Lee. "Hide-and-Seek: Forcing a Network to Be Meticulous for Weakly-Supervised Object and Action Localization." *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, doi:10.1109/iccv.2017.381.

[2] Zhou, Bolei, et al. "Learning Deep Features for Discriminative Localization." *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, doi:10.1109/cvpr.2016.319.

Data

Methods	GT-known Loc	Top-1 Loc	Top-1 Clas
AlexNet-GAP [61]	54.90^2	36.25	**60.23**
AlexNet-HaS-16	57.86	36.77	57.97
AlexNet-HaS-32	**58.75**	37.33	57.94
AlexNet-HaS-44	58.55	37.54	58.10
AlexNet-HaS-56	58.43	37.34	58.13
AlexNet-HaS-Mixed	58.68	**37.65**	58.68
GoogLeNet-GAP [61]	58.41^2	43.60	**71.95**
GoogLeNet-HaS-16	59.83	44.62	70.49
GoogLeNet-HaS-32	**60.29**	**45.21**	70.70
GoogLeNet-HaS-44	60.11	44.75	70.34
GoogLeNet-HaS-56	59.93	44.78	70.37

Table 1: Data displaying three measures of performance (described in discussion) which depict the improvement of localization with HaS applied with different patch sizes [1]

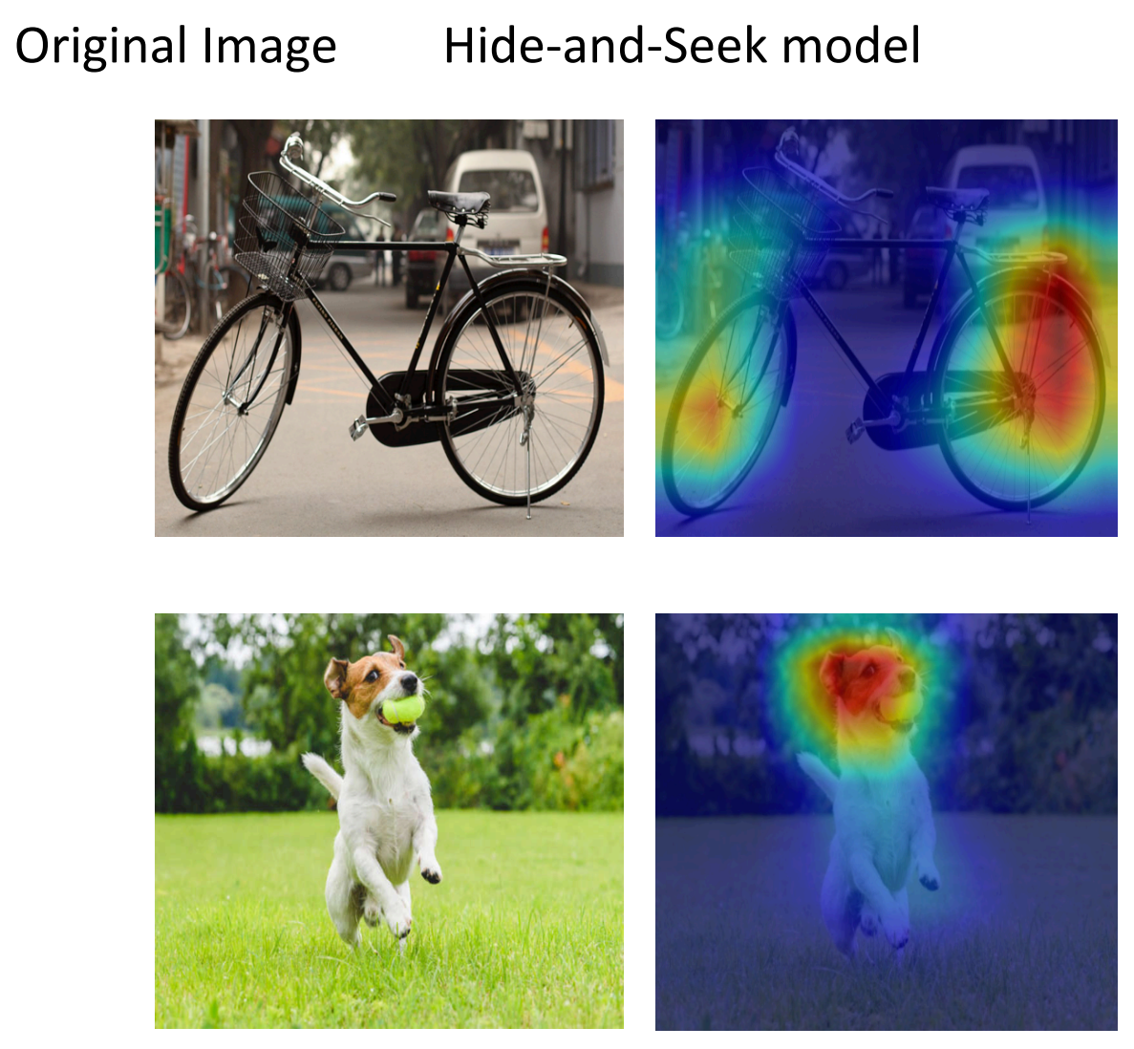
Discussion

- Due to lack of time, quantitative data was taken from the Hide-and-Seek published paper; however, the code produced in this research performs the same task
- Performance of model was assessed on three criteria:
  1. GT-known Loc = Fraction of images that have at least 50% IoU (intersection over Union) with the corresponding ground-truth bounding box
  2. Top-1 Loc = Fraction of images that are classified as the ground-truth classification AND have at least 50% IoU with ground-truth bounding box
  3. Top-1 Clas = Fraction of images that are classified as ground-truth classification (regardless of localization performance)
- It is clear from **Table 1** that when the Hide-and-Seek was applied over the AlexNet and the GoogLeNet networks, the GT-known Loc increased along with the Top-1 Loc.
- The increase in numbers remained true for every value of those two measurements
- AlexNet and GoogLeNet did have a higher value for Top-1 Clas; however, the goal of the Hide-and-Seek model was to improve localization performance more than classification
- In addition, another network is trained (AlexNet-HaS-Mixed) which consists of mixed patch sizes→ patch size is chosen randomly from 16, 33, 44, and 56, as well as no hiding (full image) → Best results were observed
- When users upload image in browser, they have options for which model to use and what class (or just top prediction) to generate CAM for

Qualitative Analysis

- As shown in the qualitative results, the Hide-and-Seek network was able to group and connect, on average, more components of an object of an image
- This increased localization would allow for more areas of an image (not just the most discriminative region) to be taken into account during detection
- For instance, HaS focuses on the face and the entire body of the rabbit (first row), whereas AlexNet only focuses on the face region → bounding box would be much more accurate with HaS than with AlexNet
- Furthermore, stronger localization will lead to more accurate bounding boxes that encompass the whole object instead of just a part of it

Failure Cases



Conclusion and Future Work

- By analyzing the failure cases, we can see that in certain angles and images, Hide-and-Seek model fails to localize most of the object, which would result in sub-optimal bounding box predictions
- Through the use of Class Activation Maps and the Hide-and-Seek deep neural network, we were able to visualize the localization performance of two different models as well as better understand the improvement in performance
- By enhancing the network, Hide-and-Seek offers many new opportunities to use weakly-supervision to train networks with voluminous datasets (millions of images) with less expense on time and human effort
- After writing the Hide-and-Seek model into PyTorch and allowing users to input data for training, it is clear that this program developed can assist programmers around the world to apply their own data to a state-of-the-art network
- Soon to be made publicly available for use to enhance the accuracy and overall performance of a detector's classification and localization
- We plan to allow the web application to access a webcam to continuously take images, which are used as frames, to form a near real-time CAM generator

Acknowledgements

I acknowledge **Krishna Kumar Singh**, my mentor, who provided me guidance throughout this extensive research project. In addition, I acknowledge my advisor, **Professor Yong Jae Lee**, who continuously provided me with feedback and advice.

I would like to thank great opportunity of the Aggie Mentors, and the entire **Aggie Mentors program**.