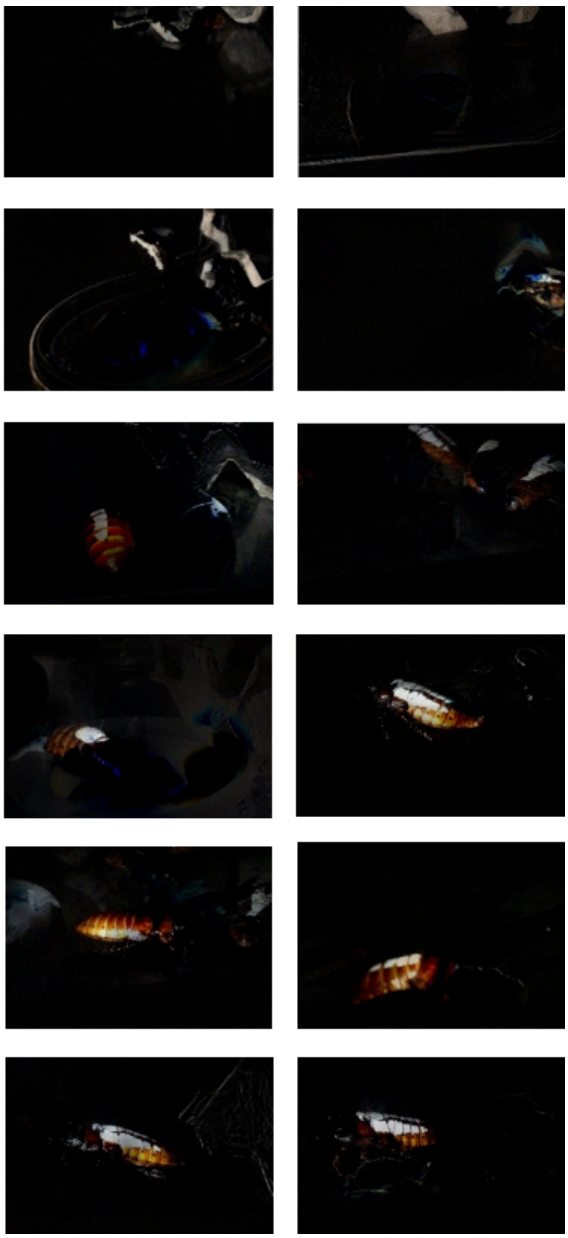
Appendix A

Original vs. Subtracted Images

As you can see, the original images have many more objects and are much more complicated than the subtracted images. When I trained the neural network with original images it wasn't able to identify any key features, and so it never detected any cockroaches. The subtraction helps black out most of the "distracting" background objects, so that the network can learn the cockroaches' key features properly and therefore perform better.

Original Images

Subtracted Images



Appendix B

Network Performance Figure of Merit

To evaluate the network's performance, I developed a figure of merit. There are two type of errors that it could make, it could miss detecting a cockroach or it could falsely detect a cockroach. Since all positive images, images with cockroaches, have the potential to be misses, miss errors contribute $\frac{\text{total misses}}{\text{total positives}}$. Since all negative images, images without cockroaches, have the potential to be false hits, false hit errors contribute $\frac{\text{total false hits}}{\text{total negatives}}$. Thus, we use a figure of merit of $1 - \frac{\text{total misses}}{\text{total positives}} - \frac{\text{total false hits}}{\text{total negatives}}$. The closer the FoM is to 1, the better it performs.

Appendix C

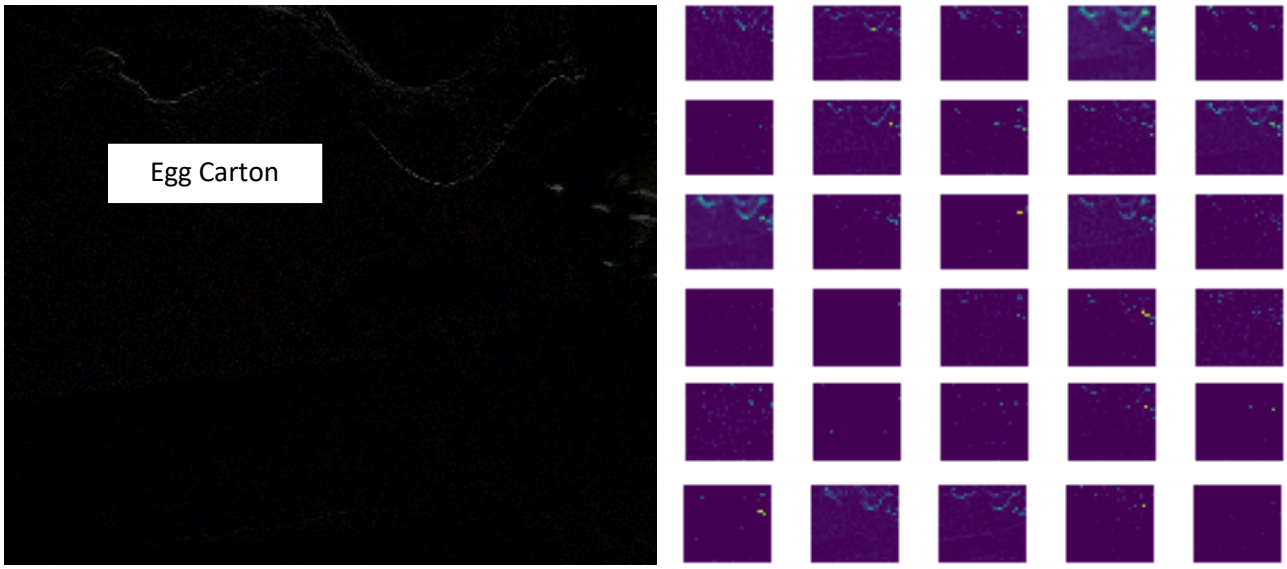
Example of a True Positive's Heat Maps for 30 Neurons (Total of 140 Heat Maps per Image)

These heat maps and the features highlighted/acknowledged in them are combined with a weighted sum. The features highlighted/acknowledged by the nodes with the largest weights and biases are used as key features for the detection process, so not all nodes need not highlight the key feature for a cockroach to be detected. Most of the heat maps of this true positive highlight/acknowledge the cockroach's color and shape.

Appendix D

Example of a False Hit's Heat Maps for 30 Neurons (Total of 140 Heat Maps per Image)

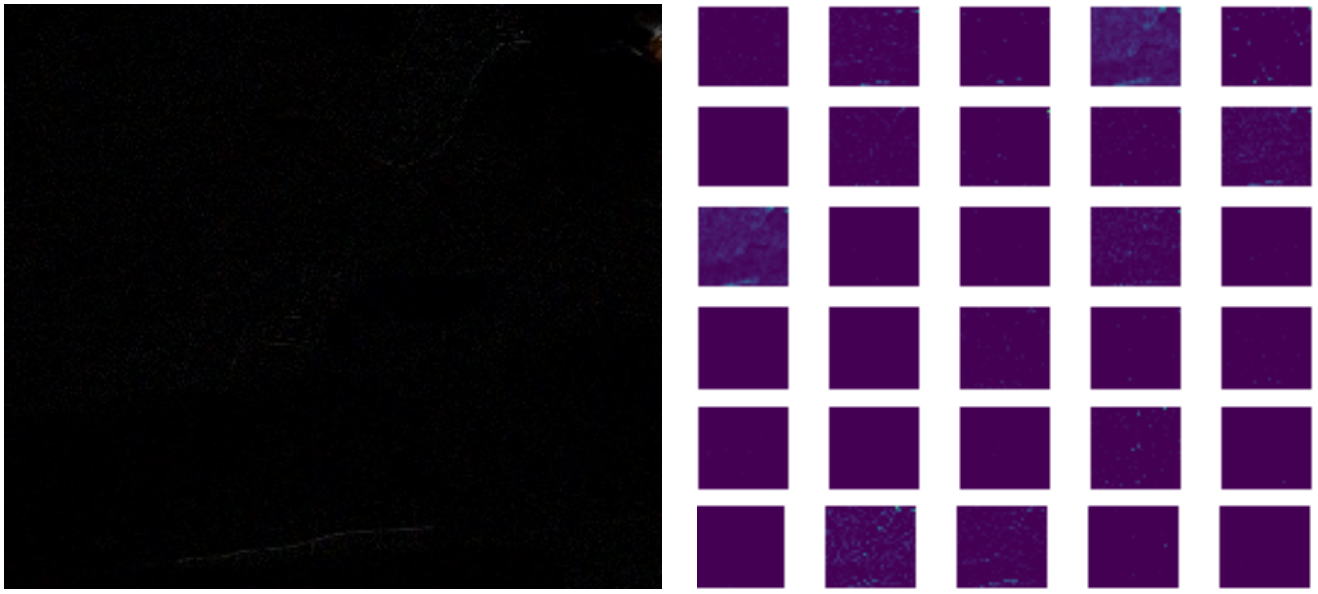
These heat maps and the features highlighted/acknowledged in them are combined with a weighted sum. The features highlighted/acknowledged by the nodes with the largest weights and biases are used as key features for the detection process, so not all nodes need not highlight the key feature for a cockroach to be detected. Most of the heat maps of this false hit highlight/acknowledge the egg carton's cockroach-like shape.



Appendix E

Example of a True Negative's Heat Maps for 30 Neurons (Total of 140 Heat Maps per Image)

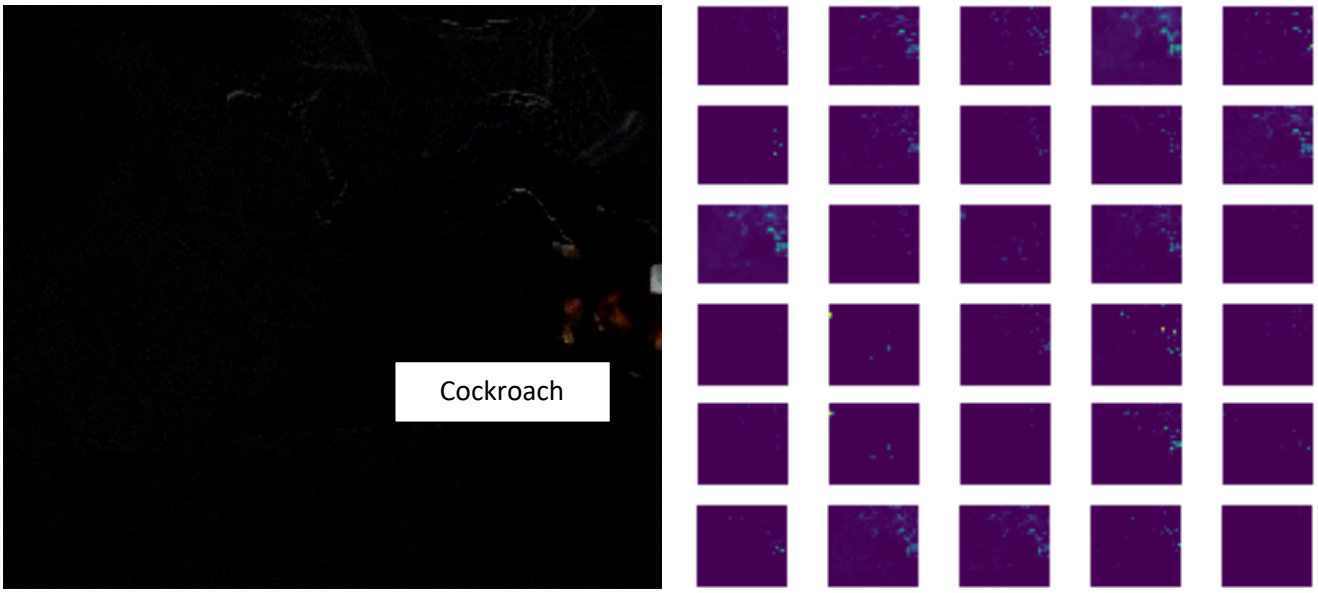
These heat maps and the features highlighted/acknowledged in them are combined with a weighted sum. The features highlighted/acknowledged by the nodes with the largest weights and biases are used as key features for the detection process, so not all nodes need not highlight the key feature for a cockroach to be detected. None of the heat maps of this true negative highlight/acknowledge any clear patterns.



Appendix F

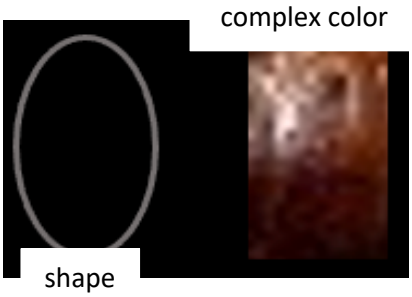
Example of a Miss's Heat Maps for 30 Neurons (Total of 140 Heat Maps per Image)

These heat maps and the features highlighted/acknowledged in them are combined with a weighted sum. The features highlighted/acknowledged by the nodes with the largest weights and biases are used as key features for the detection process, so not all nodes need not highlight the key feature for a cockroach to be detected. Most of the heat maps of this miss highlight/acknowledge the cockroach's color.



Appendix G

Verification Image's Heat Maps (Complex Brown)



Appendix H

Resources Used for Cockroach Detecting Network

I programmed the Keras neural network in Python and ran it in Google Colaboratory, a free Google service with free GPU access. I needed to use these GPUs to keep the network's training time reasonable even with the large number of images and large number of epochs.