Final Project Report

Scott Watkins, Joshua Williams, Dong Guo, and Lawrence Lewis

A. Source of CNN Implementation:

Matlab's "Deep Learning Toolbox" was implemented in this project.

B. Information about Experiment:

Input Data Information

	Input Image Size [pixels]	Number of Classes	Number of Images Per Class	Training Set Size
Trial 1	150X150	2	2872	90%
Trial 2	150X150	3	1180	90%
Trial 3	150X150	4	120	90%

Hyperparameters (Identical for all Trials)

- Number of Epochs: 50
    - Accuracy did not increase when number of epochs were increased
- Initial Learning Rate: 0.01
    - Reduced by 90% per epoch
- Momentum Value: 0.9
    - Used Stochastic GD with Momentum
- Batch Size: 256
    - Larger batch sizes were tested, no improvements in accuracy were made.

Summary of How Input Was Conceived

Using Dr. Sarraf's feature detection program, we collected images that included worms and non-worms. We then manually labeled them and classified them by number of worms in the

image. The images were padded (with zeros) to a size of 150 by 150 pixels after first screening the initial size. Images that were deemed much too small or much too large were discarded. We increased the amount of images in our training data by rotating the images we currently had. This increased our number of training images by a factor of four. These images were then saved in a format that the Deep Learning Toolbox could use. We then further separated the images into a testing and training set. The hyperparameters used during training with these images are discussed above. The training and test sets were randomly generated from the set of images, 90% were used for training and the other 10% were used for testing.

C. Classifier Performance

Trial 1

Training Error:	0.01
Test Error:	0.42
Training accuracy:	100%
Test accuracy:	87.8%

Trial 2

Training Error:	0.09
Test Error:	0.8
Training accuracy:	100%
Test accuracy:	75.4%

Trial 3

Training Error:	0.02
Test Error:	1.18
Training accuracy:	100%
Test accuracy:	60.4%

D. Training and Test Times

All results discussed were performed using MATLAB's Parallel Processing Toolbox on an NVIDIA GeForce 1050Ti GPU. Performing these calculations took orders of magnitude longer on a CPU.

By nature of MATLAB's CNN, the system calculates test accuracy every 5 iterations. This means the training time includes test time multiple times; however, test time is much smaller compared to training time so its effect on runtime is negligible.

Trial 1

Training Time: 13 min 7 s
Test Time: 11.5 s

Trial 2

Training Time: 5 min 14 s
Test Time: 6.9 s

Trial 3

Training Time: 1 min 45 s
Test Time: 1.8 s

E. Comparison of Performance to Logistic Regression Classifier

All comparisons use **Trial 1** from CNN implementation because **Trial 1** used two classes, and the Logistic Regression implementation also used two classes.

- Accuracy
  - CNN: **87.8%**
  - Logistic Regression: **67.5%**
- Runtime
  - CNN Training: **13 min 7 s**
  - CNN Test: **11.5 s**
  - Logistic Regression Training: **2 min 51 s**
  - Logistic Regression Test: **0.3 s**