The Relationship Between Memory Performance and Hippocampal Volume: A Machine Learning Model Using Deep Learning

Michael Popa (map21@rice.edu)
Department of Cognitive Sciences, Rice University, Houston, TX 77005 USA

Abstract

Human memory has been shown to rely on various neuroanatomical regions of the brain, with the most well-researched being the hippocampus, a region deep within the medial temporal lobe. However, the relationship between memory performance and hippocampal volume has been an area of much debate within the field of neuroscience, as studies tend to produce mixed results. To explore this issue, we put forth a neurocomputational model simulating the various subregions of the medial temporal lobe and hippocampus. This is done through deep learning with a focus on feedforward and convolutional autoencoders in Python using Keras with Tensorflow. The layers will be varied in dimensionality, and the output from the autoencoder will be interpreted based on these changes. The model will be trained and tested on images from the Fashion MNIST dataset, which represent images of clothing. Our findings suggest that hippocampal volume does play a role in memory performance, depending heavily on the specific layer that is damaged. In addition, our model shows that the ability to generalize a visual stimulus does not come solely from the shape and structure but may come from the shade of the image as well.

Keywords: Medial Temporal Lobe Simulation; Autoencoder; Convolution; Memory; Hippocampus

Introduction

The performance of human memory relies of various neuroanatomical regions in the brain. Out of them, the most well-researched is the medial temporal lobe (MTL). Specifically, a region deep within the medial temporal lobe called the hippocampus. This region has been proven to be critical for long-term memory storage, as well as retrieval. Although the literature shows that the content of memories are not necessarily kept in the hippocampus, we know from a variety of famous amnesia cases that this region is at least necessary for encoding and retrieving memories. Nonetheless, one major area of debate within memory research is whether or not the size of the hippocampus can influence the performance of our memories. This question is valuable within the field, as individuals with neurodegenerative diseases, as well as certain neurological disorders have varied hippocampal volumes. For instance, those with Alzheimer's and post-traumatic stress disorder (PTSD) have shown to have reduced hippocampal volume, while individuals with autism spectrum disorder (ASD) demonstrate the opposite. As for differences in performance, some evidence shows that greater hippocampus volume is associated with better memory function, such as in the laboratory task. A 2014 study by Pohlack et al. found a positive correlation between memory measures and percent of hippocampal volume. We also know those with Alzheimer's have much worse memory performance. So, although these findings (combined with a degree of intuition) may push one to believe that worse memory performance is associated with reduced hippocampal volume, the literature also shows evidence that the correlation is not as strong as once believed. It was quite common for scientists to accept this correlation in earlier literature, until a 2004 study by Dr. Cyma Van Petten (Binghamton University) found that size does not matter for memory in older adults. A more recent and separate 2020 study by Clark et al. supported Dr. Petten's findings, claiming little evidence that hippocampal gray matter volume was related to task performance.

An important note for this relationship is the impact of experience on hippocampal volume and memory performance. One of the most famous cases of this comes from the London taxi-cab driver experiment. In this study, drivers had to memorize a large amount of information (the map of London) in order to pass their license test. In doing so, the study found that drivers had larger hippocampi than the normal population. An intuitive explanation could be that their hippocampus grows in direct relation to the amount of information it needs to encode and store. A follow-up study proves that changes in hippocampus size follow the amount of studying necessary to pass the test.

When investigating the memory performance of those with neurological disorders and diseases, we have some interesting findings as well. For instance, the hippocampal volume reductions found in individuals with PTSD do not show any corresponding changes in long-term memory function. The same goes for those with anxiety. However, as mentioned earlier, individuals with ASD have a significantly higher hippocampus volume. And although this size difference is noteworthy, what is remarkable is that there is no accompanied benefit (or drawback) on memory performance. In fact, a 2021 paper by Fyfe et al. found that the prevalence of amnesic dementia is four times higher in individuals with ASD. This means that those with ASD are four times more likely to develop the symptoms common in those with amnesia. Intuitively, if increased hippocampal volume was related to memory performance, the prevalence rate of amnesic dementia in ASD individuals should not be that high.

Given these findings, the main question that this paper seeks to answer is whether or not the size of hippocampus has a direct impact on memory performance.

Computational Framework

Clearly, the findings of hippocampal volume in relation to memory performance are mixed. To further investigate this

research question, a computational framework will be used to simulate various regions of the medial temporal lobe, including the subregions of the hippocampus. In particular, the regions will include: (1) the temporal cortex, which receives input from various cortical regions, (2) the entorhinal cortex, which receives the input from the temporal cortex and is the main input region of the hippocampus, (3) the dentate gyrus (DG), which receives information from the entorhinal cortex through the performant pathway and is generally responsible for pattern separation, (4) the CA3, which receives information from the DG through the mossy fiber pathway and is mostly responsible for pattern completion but can support pattern separation as well, and (5) the CA1, which receives information from the CA3 through the Schaffer collaterals and is the main output back to the entorhinal cortex. Pattern separation refers to the ability to distinguish between similar but different objects, places, or contexts. In other words, it allows us to discriminate between stimuli. Pattern completion refers to the holistic retrieval of experiences given a cue. In other words, it allows for the reconstruction of a memory given partial or degraded cues and allows us to generalize. These three regions, the DG, CA3, and CA1 are known as the trisynaptic circuit. Once these three process the information, it is sent from the entorhinal cortex back to the temporal cortex, where it is then sent to other cortical areas for further processing. It's important to note that the CA2 is not included due to its limited research into its role in memory formation. The literature suggests the CA2 is mostly important for social memory, but that is not important for our research question at hand.

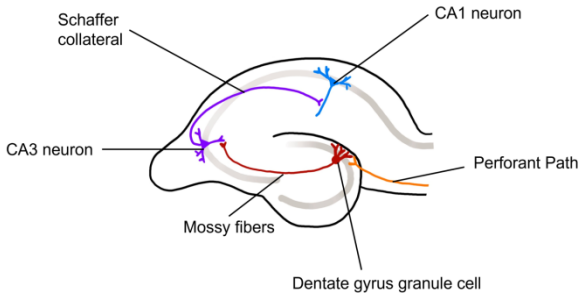

Figure 1: The trisynaptic circuit of the hippocampus

To make this model possible, there will be a focus on the Multiple Trace Theory, which states that memories are a collection of traces, each corresponding to a specific episode in which the memory's contents were encoded.

This model will attempt to understand how the output changes in relation to differences in parameters. We will use a deep neural network, specifically, an autoencoder in which each layer is a region within the MTL. The reason why we use an autoencoder comes from its similarity to how memories are processed within the hippocampus. When a memory is encoded, not all information is retained. The input (the specific stimulus) will be heavily compressed, and so only the most important and prominent characteristics of that stimulus will be remembered. This is why when we recall a memory, not all details will be remembered. This is crucial for efficient storage. Autoencoders work in a similar manner, where the high-dimensional information gets compressed in a lower-dimensional form, meaning that there must be a retainment of the most important information. It is then reconstructed based on the compressed form as accurately as possible. This study will be done through the lens of Bartlett's theory of reconstructive memory, where memory is an inherently fallible system that is reconstructive in nature. In other words, when we recall a memory, we essentially reconstruct it by filling in each gap with whatever information is available to us, which will almost always inadvertently introduce inaccuracies to the memory. We hope that through this simulation model, we can get a better understanding of how MTL size can impact memory performance.

Methods

Fashion MNIST Dataset

Our model was trained through the Fashion MNIST dataset, which consists of 70,000 28x28 black and white images from 10 clothing categories. Of this 70,000, the first 60,000 are typically used for training with the remaining 10,000 being used to test the model's performance. As this is a very large amount of data, we will only use a subset of this 70,000. Specifically, we used only 4,000 images/stimuli for computational efficiency. To understand the effects of stimuli frequency, we varied the number of times each stimulus appears in the training set. Concretely, of these 4,000: The first 1,000 were repeated only once (1000 x 1), the following 100 were repeated ten times (100 x 10), the next 10 were repeated one hundred times (10 x 100), and only 1 image was repeated one thousand times (1 x 1000). Adding these up, we got our set of 4,000 images that was used for training.

Figure 2: Examples of clothing images

Implementation

The entire process was implemented in Python version 3.12.0 in Google Colab. The original implementation was in Jupyter Notebook, but Google Colab was chosen for

computational efficiency. For our deep learning model, we opted to use TensorFlow with Keras, while our visualizations and numerical computations came from Matplotlib and NumPy respectively. Additionally, convolutional layers with Max Pooling and UpSampling were introduced as well but with limited use.

Preprocessing

There was only one main preprocessing step to make our data easier to work with. As we were working with image data, the values representing each image typically span from 0, meaning black, to 255, meaning white. Each pixel is represented by one byte, or 8 bits of data. To deal with this, we normalized the pixel values to a range of 0 to 1 for both the training and test sets by diving the data values by 255.

Modeling

As mentioned earlier, the simulation of these neuroanatomical regions was done using an autoencoder, which each layer of the neural network representing a specific region of the MTL. Our network was divided into two sequential models, the encoder and decoder. The encoder has four layers, the first representing the temporal cortex, having 784 neurons, the second being the entorhinal cortex, having 512 neurons, the third representing the DG, having 384 neurons, and lastly, the fourth layer representing the CA3 which has the least amount of neurons, 256. In the encoder, each layer has a ReLU activation function (to avoid the potential of a vanishing gradient). The input into the encoder is a 2D 28x28 shaped image that is flattened into a 1D array of size 784 (since 28 x 28 = 784). This is why the temporal cortex has 784 neurons.

Our decoder had only three layers, the first representing the CA1 with 384 neurons, the second representing the entorhinal cortex with 512 neurons, and lastly, the temporal cortex which had 784 neurons. The first two layers of our decoder had a ReLU activation function, while our output layer was implemented using a sigmoid activation. The output is then reshaped back into a 2D 28x28 shaped image.

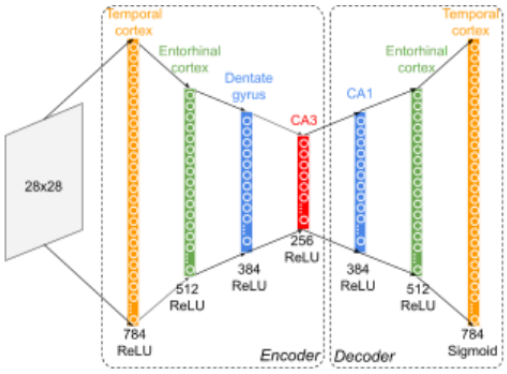

Figure 3: Autoencoder architecture

To capture the most prominent features from each image, we used a separate autoencoder that introduced convolutional layers, as well as Max Pooling and UpSampling. This is called a convolutional autoencoder, which is designed for feature learning and dimensionality reduction. The convolutional layers apply a convolution operation on the input, essentially passing it through a set of learnable filters. Max Pooling then reduces the spatial dimensions, extracting the maximum value from each patch of the feature map. Up Sampling is useful for the decoder, as it increases the spatial dimensions of the learned compressed representation to reconstruct the original input. Doing so, we learn the fine-grained features.

The model was trained on all training set items for a varying number of epochs. Our 'control' autoencoder was only trained for ten consecutive epochs, but our modified autoencoder (with changes in the number of neurons per layer) was trained for around 50 - 100 epochs. The reason we increase the number of epochs is to understand how long it would take to return an accurate output for a lesioned hippocampus. Our optimizer was stochastic gradient descent with adaptive moment estimation (the Adam optimizer in Keras). Additionally, we used a loss function to modify weights via backpropagation of mean squared error recall.

To introduce sparseness, our combined loss function included two terms:

$$L = \sum_{i,o}^{N} (y_i - y_o)^2/N + \lambda \sum_{h \in CA3} |y_h|$$

The first term represents the accuracy cost of network recall, and is the mean squared difference between each input and corresponding output neuron. The second term is the resource cost, representing the sum of activation of each hippocampal neuron in the CA3 layer. The reason we focus on the CA3 layer is because it is the final layer of our encoder, so our penalty cost only interests the neurons in this layer. Our hyperparameter lambda is the regularization parameter, representing the weight of the penalty. The resource cost penalty is equivalent to the L1 penalty used in Lasso regression. Our L1 regularization parameter will be around ~0.00001. Doing so allows certain features to be forced to zero, in essence, deactivating connections between neurons within our neural network. This is done to encourage sparseness; retain the most critical features of our input. Using the error squared can help us see exactly what is lost from the stimulus during memory recall.

Figure 4: Example of detail that is lost during recall

The idea of our model was to understand how similar the output is to the input given modifications to our parameters.

Since our main question was whether the size or volume of the hippocampus makes a difference in memory processes, varied the number of neurons within the layers of the autoencoder to stimulate hippocampal lesions. We sought to analyze how much a reduction in the layers of the neurons impacts the output to the model. By taking into account stimulus frequency, we hoped to understand how much the model would bias to generalize to the most frequent stimulus' characteristics. In other words, how much would the model generalize the lower frequency images, prioritizing getting its overall gist over any potential details.

Results

To better understand the output of our model, we will divide this section into two subsections, one dealing with a control or 'healthy' autoencoder with the aforementioned number of neurons, while the other deals with the modifications to the number of neurons per layer, or the lesions. In all instances, the model randomly picks 8 images for each epoch and provides and output that is meant to represent the recalled image.

Control Autoencoder

To understand what the autoencoder does every epoch, we first printed 10 epochs of the same 8 stimuli/images over each epoch. In the first epoch, there is no output, as the input just got fed into the model, so it first has to learn the features. With every epoch, we can slowly begin to see a general shape emerge that matches the input. After around 10 epochs, we begin to see some level of stabilization, where the input (generally) matches the output.

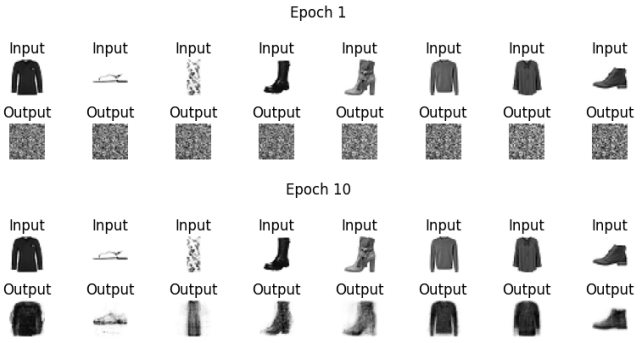

Figure 5: An example of a regular autoencoder at epoch 1 and epoch 10 (no L1 penalty)

Even without model sparseness (the L1 penalty), we see in Figure 5 that the neural network naturally learns to prioritize gist over detailed learning. The overall shape is captured much more often than any particular details. We can especially see this in the third image from the left, there the shape of the dress is kept over the details of the dress. Figure 5 is just an example of how our model out will look like, but it does not take into account stimulus frequency.

Using convolutional layers, we understand what the most prominent features from each image are. We see there is a bias towards the shape/structure of the item rather than any particular characteristics. In other words, there is a prioritization of getting the overall gist of the image rather than any specific details within the object (see Figure 6).

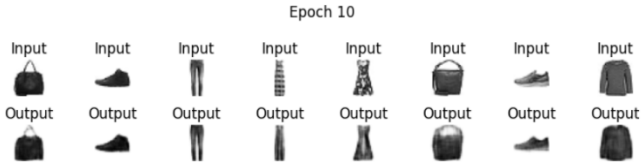

Figure 6: Convolution autoencoder output after 10 epochs

The convolutional autoencoder output was not massively different from our feedforward neural network autoencoder with simple sequential models. Therefore, convolutional autoencoders were not used for the rest of the study.

Taking into account stimulus frequency, we can begin to understand how and why the model learns certain features over others. Specifically, stimuli that are more frequent within the data that the model is being trained on will cause a massive bias to generalize to that specific stimulus' characteristics. In other words, the lower frequency images will look more like the higher frequency images. To understand this better, we can analyze Figure 7:

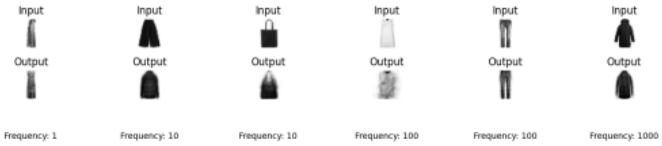

Figure 7: Epoch 93 of a control autoencoder taking into account stimulus frequency

Note: we have a total of 8 examples per epoch, meaning two clothing examples per frequency, but most figures will have fewer for readability purposes. In Figure 7, we can see the most frequent stimulus to be the black coat. Because of this, the neural network prioritizes the black coat by strengthening the weights responsible for that image, causing the weights for lower frequent stimuli to be less represented. So, in this figure we can see the effects of this. The second and third image are biased much more towards a black coat than any other image. In the second image, the input was a black pair of shorts, but we can clearly see the output matches that of the most frequent stimulus. In the third image, the input image is a black purse, but the output has the slight shape the coat. In both cases, it seems to be the case that once the model understands the input image is black, the model will have high confidence in assuming it is a coat. This does not happen with the other stimuli, as those images tend to be a bit grayer and whiter. What this may tell us is that the shade of an image plays a massive role in determining how the bias occurs. Particularly, the most prominent characteristic in determining what the model prioritizes is the shade of the

stimulus. In addition, it appears that the amount of time and training required for these images does not make a significant difference, as this was after 90+ epochs of consecutive training that the model underwent.

Lesioned Autoencoder

To understand the impact each layer has on the model, we will essentially 'damage' each layer by reducing the number of active neurons for a specific layer. To do so, we bring down the number of neurons to around 10 neurons for a layer, reducing its usefulness.

If we start by damaging the temporal cortex, we can understand the major change that occurs in model output. Specifically, the number of epochs the autoencoder takes to return a valid output increases drastically. In our control autoencoder, we see that it takes around 10 epochs to get an output that similarly matches the input. When damaging the temporal cortex however, we see that number jump to around ~50 epochs (see Figure 8).

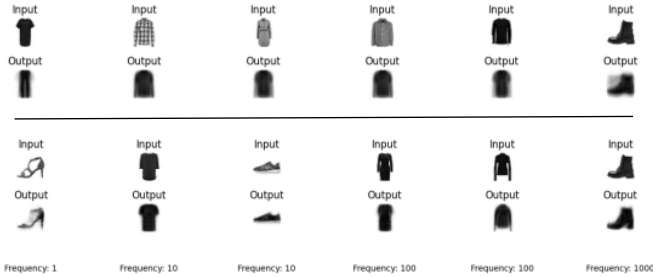

Figure 8: Temporal cortex lesion output at epoch 10 (top) compared to epoch 49 (bottom)

As we see in Figure 8, certain details of clothing begin to appear as the number of epochs increases. The heels have a shape that is closer to the input, and the sneakers have a white sole at the bottom, matching the input. Additionally, the images in epoch 10 tend to be a bit grainy and noisy, which is not present at epoch 49. So, performance of a lesioned temporal cortex tends to become similar to that of a control MTL, but requires more training to occur. This suggests damage to the input of the MTL will require more time for a memory to stabilize (depending on the level of damage).

One of the most interesting findings comes from the damage to the CA3 layer, the innermost layer of the encoder. From here we have the L1 penalty that is applied at the end of the encoder, encouraging the storage of the most important features from the input before it gets decoded. Recall, the CA3 layer is most important for pattern completion but can perform pattern separation as well. The image data will end up being compressed to 10 neurons, so the decoder will only have 10 neurons to pull from.

After around ten epochs, we see some level of similarity with the temporal cortex lesion, but there is one crucial difference. What is new is that the model begins to struggle to classify the image. In other words, it cannot confidently differentiate between clothing items; there is confusion on what exactly to encode and/or decode.

We can see this in Figure 9:

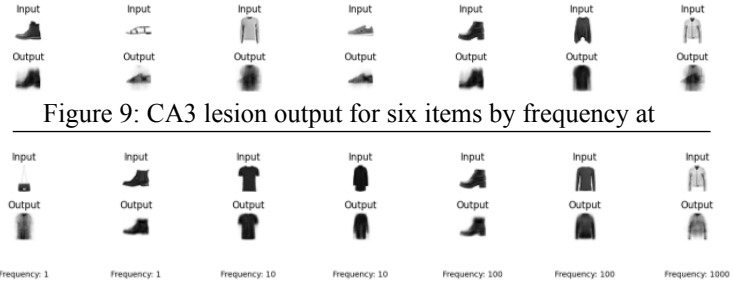

Figure 9: CA3 lesion output for six items by frequency at epoch 10 (top) and epoch 50 (bottom)

In Figure 9, we can see a clear example of what is happening. The first item, the one after 50 epochs, is completely misclassified. There are little to no similarities between the input (purse) and output (shirt), implying the damage done to this layer of the hippocampus caused the model to classify the purse as a shirt. This is a unique deficit that has not been seen yet, as other instances of misclassification occurred when there was some level of similarity (such as the shade or general shape). Here, there is little to no similarity between the input and output.

What is even more remarkable is the output of the most frequent stimulus. Specifically, how frequent the stimulus appears in the dataset does not seem to yield a significant difference in misclassified output.

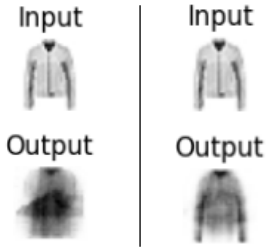

Figure 10: CA3 lesion output for the most frequent stimulus at epoch 10 (left) and epoch 50 (right)

We can see in Figure 10 the remnants of shoe within the shirt in both instances. What this tells us is there exists model confusion between two completely independent and unique stimuli that does not seem to be strongly influenced by the number of epochs. This is unique, as in Figure 9 we can see less frequent stimuli being correctly classified, while Figure 10 shows us that the most frequent stimulus is not being classified properly or confidently.

Lastly, the dentate gyrus was damaged, again bringing down its number of neurons to about ~10. However, in doing so, the output became completely non-existent, regardless of the number of epochs. To fix this, we opted to use around ~50 neurons instead of 10. Recall, the number of original neurons in the dentate gyrus layer was 384 neurons.

What we see now is a mix of the prior two lesion outputs. After 10 epochs, we can see the model tends to generalize the lower frequency stimulus to the high frequency stimulus. In other words, the lower frequency stimuli look like the highest frequency stimulus. As the number of epochs increases, we begin to see a high level of stabilization, and we notice outputs similar to the control autoencoder's output.

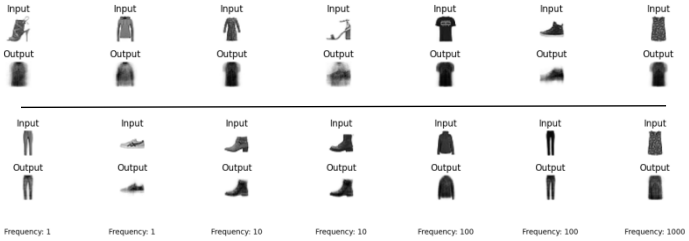

Figure 11: DG lesion output at epoch 10 (top) and epoch 50 (bottom)

In Figure 11, we see some degree of classification confusion, and there is a similar pattern of generalization bias to the most frequent stimulus at epoch 10. However, at epoch 50, we see that the shape and overall structure of the input image is correctly recalled, while the shade is not. Specifically, in image two of epoch 50, we can see that the sneaker is recalled as black rather than gray, which goes back to what we found in the control autoencoder.

Discussion

This study sought to understand how memory recall performance changes based on lesions done to different layers. The most remarkable finding was the lesioned CA3 layer. The CA3 is known to be responsible for pattern completion but can also perform pattern separation to a lesser extent. The output showed that classifying the input involved much confusion, even after 50 epochs. The most intuitive explanation is that the model failed to pattern separate between the input stimuli. However, considering its role in pattern completion, a separate explanation could be that pattern completing to a specific stimulus failed, therefore resulting in a completely different stimulus being picked for the output. This question is intriguing and would require further research.

Additionally, based on our results from the control autoencoder, it seems that the shade of the clothing plays a massive role in determining generalizability. We saw this when the input images with similar shades to the most frequent stimulus had output that matched that of the most frequent stimulus. Further research may look at what other characteristics tend to be learned the most once the shape or structure has been learned.

Overall, this paper confirms that the size of the hippocampus does in fact impact memory performance, but the deficit depends on the specific layer of the region that is losing volume. In all three lesion examples, we do end up getting recalled output, and the output for the most part tends to match the input. It is in the details, as well as the amount of training the model undergoes where we begin to see significant differences. It's important to also note that the aforementioned studies that provided mixed findings for the relation of memory performance and hippocampal volume used different metrics to determine what is high or low memory performance accuracy. The memory tasks used in the studies are varied, testing different aspects of memory rather than just the recall of images.

Further work on this model could focus on damaging all layers equally, or working on tweaking the optimizer and loss function differently. Also, the L1 penalty being tested at different values can help us understand how much neuronal sparseness is necessary to encode and decode a particular stimulus representation. Lastly, a new model may look at different aspects of memory, such as synaptic plasticity, as well as different sensory modalities apart from vision.

Acknowledgments

The guidance and support of Dr. Bartlett Moore IV (Rice University), as well as Annlyle Diokno (Rice University) was a source of inspiration throughout the project that led to the completion of this research. Their feedback, patience, and assistance proved to be invaluable, and this work could not be done without them.

References

Stocco, A., Smith, B. M., Leonard, B., & Hake, H. S. (2023). *Efficient Memory Encoding Explains the Interactions between Hippocampus Size, Individual Experience, and Clinical Outcomes: A Computational Model*. https://doi.org/10.1101/2023.11.22.568352

Van Petten C. (2004). Relationship between hippocampal volume and memory ability in healthy individuals across the lifespan: review and meta-analysis. *Neuropsychologia* 42, 1394–1413. 10.1016/j.neuropsychologia.2004.04.006

Maguire, E. A., Gadian, D. G., Johnsrude, I. S., Good, C.D., Ashburner, J., Frackowiak, R. S., & Frith, C. D.(2000). Navigation-related structural change in the hippocampi of taxi drivers. *PNAS*, 97(8), 4398-4403.

Maguire, E. A., Woollett, K., & Spiers, H. J. (2006). London taxi drivers and bus drivers: a structural MRI and neuropsychological analysis. *Hippocampus*, 16(12),1091-1101.

Pohlack, S. T., Meyer, P., Cacciaglia, R., Liebscher, C.,Ridder, S., & Flor, H. (2014). Bigger is better! Hippocampal volume and declarative memory performance in healthy young men. *Brain Structure and Function*, 219, 255-267.

Clark, I. A., Monk, A. M., Hotchin, V., Pizzamiglio, G., Liefgreen, A., Callaghan, M. F., & Maguire, E. A. (2020). Does hippocampal volume explain performance differences on hippocampal-dependant tasks? *NeuroImage*, *221*, 117211. https://doi.org/10.1016/j.neuroimage.2020.117211