



Investigating Intra-Layer Neuronal Mechanisms

Mohammad Zamani
Student No. 610399135

University of Tehran- CS Department

Contents

1	Introduction	2
2	Experimental Setup	2
2.1	Network Architecture	2
2.1.1	First Architecture	2
2.1.2	Second Architecture	3
2.2	STDP Learning Rule	3
2.3	Parameter Configurations	4
2.3.1	Input Patterns	4
2.3.2	Spike Trace	5
2.3.3	STDP Parameters	5
2.3.4	Neuron Model	5
3	Lateral Inhibition	5
3.1	Different Input Pattern	6
3.2	Different Input Pattern With Intersection	7
3.3	Same Input Pattern	8
4	KWTA	9
4.1	Different Input Pattern	10
4.2	Different Input Pattern With Intersection	11
4.3	Same Input Pattern	13
5	Homeostasis	13
5.1	Homeostatic Process	13
5.2	Homeostasis	14
5.3	Purpose of Homeostasis	14
5.4	Experiments	15
5.5	Conclusion	16

1 Introduction

This project delves into the intricacies of intra-layer neuronal mechanisms and their profound effects on learning processes within spiking neural networks. Using the CoNeX framework, we systematically implement and evaluate several key mechanisms, including Lateral Inhibition, k-Winners-Take-All, and Homeostasis, to understand their roles in shaping network behavior. Our approach involves training spiking neural networks with the Spike-Timing-Dependent Plasticity (STDP) learning rule, allowing us to observe the nuanced interactions between neurons within a layer.

Through a series of controlled experiments, we assess how different configurations and parameters of these mechanisms influence learning process and network performance. Specifically, we analyze how variations in neuronal sensitivity and competitive interactions affect the extraction of salient features and the overall learning trajectory. By comparing networks with different intra-layer mechanisms, we aim to identify optimal strategies that enhance learning outcomes.

Our results provide valuable insights into the dynamics of neural interactions within a layer and their critical impact on learning processes.

2 Experimental Setup

2.1 Network Architecture

2.1.1 First Architecture

In our previous project, we discussed a spiking neural network composed of two layers: an input layer and an output layer. The input layer contains two groups of neurons, each group having four neurons with distinct input patterns. The output layer consists of a single group with two neurons. We employed a fully connected pattern, where each neuron in the input layer is connected to all neurons in the output layer, as illustrated in Figure 2.

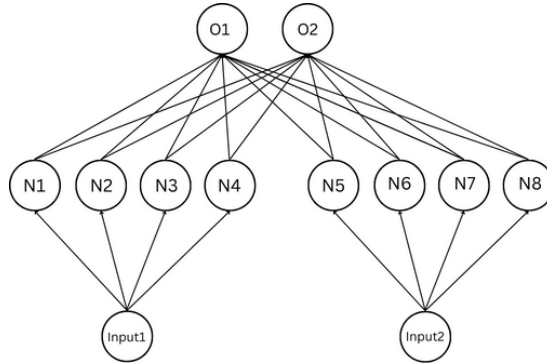


Figure 1: Network Architecture

2.1.2 Second Architecture

In this structure, we have a neuron group with five neurons in the output layer and five other neuron groups in the input layer, each consisting of four neurons. Each neuron in the input layer receives a distinct pattern as input. We used a full connectivity pattern, meaning every neuron in the input layer is connected to all five neurons in the output layer.

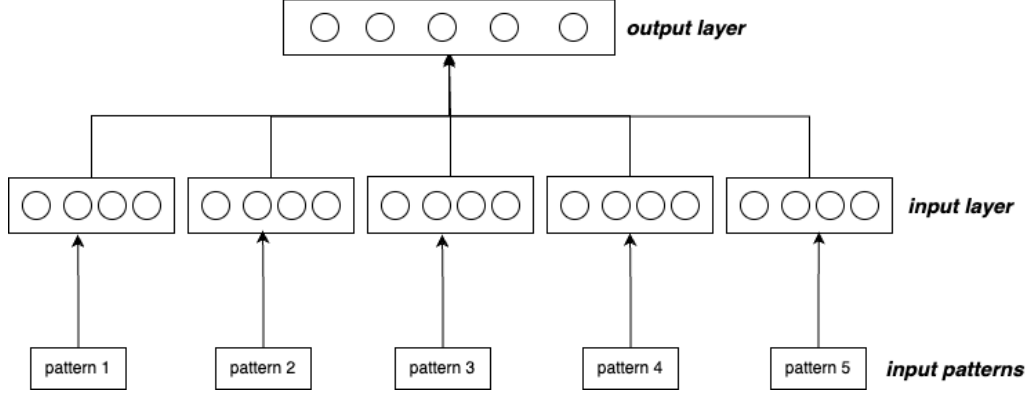


Figure 2: Network Architecture

2.2 STDP Learning Rule

In the Spike-Timing-Dependent Plasticity (STDP) process, if an input spike to a neuron typically occurs just before the neuron's output spike, the synaptic connection for that input is strengthened. Conversely, if an input spike generally occurs just after an output spike, the synaptic connection is weakened. This timing-dependent adjustment is the essence of "spike-timing-dependent plasticity." Consequently, inputs likely causing the post-synaptic neuron's excitation become more influential, while inputs that are unlikely to cause the excitation become less influential. Over time, this process refines the network by reinforcing a subset of connections while diminishing the impact of others to nearly zero. Since a neuron generates an output spike when many inputs arrive within a short time frame, the remaining subset of inputs tends to be those that are temporally correlated. Additionally, inputs that consistently precede the output spike and indicate early correlation become the primary inputs to the neuron, effectively optimizing the network for detecting and responding to relevant patterns.

If we have two electrodes placed within the neuronal membrane, and we use one electrode to stimulate the neuron and the other to measure the neuron's response, we can observe that if the previous neuron is activated and then the next connected neuron is activated shortly afterward, the synaptic strength between these two neurons increases. This indicates that the change in synaptic weight can be a function of the time difference $t_j^f - t_i^f$.

STDP

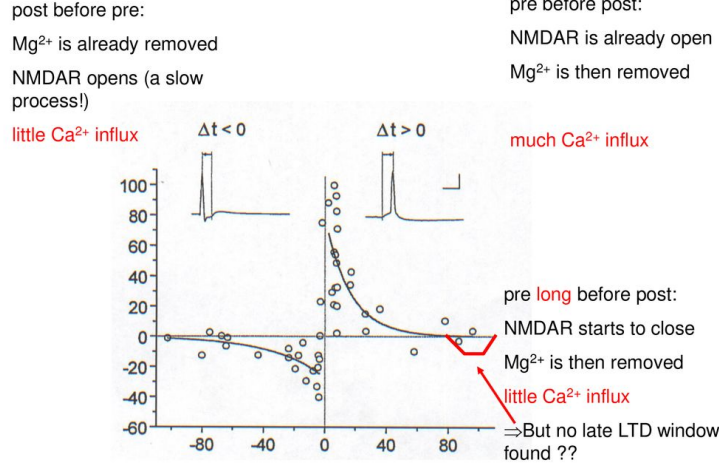


Figure 3: STDP Learning Process

LTP is the process by which synaptic strength is increased. It occurs when the presynaptic neuron fires just before the postsynaptic neuron. The closer in time these spikes occur, the stronger the synaptic connection becomes. The rationale behind LTP is that if a presynaptic neuron consistently helps to trigger a postsynaptic spike, it should be rewarded by strengthening the connection, making it easier to trigger future spikes. This process reinforces pathways that are frequently used, thereby promoting learning and memory formation.

LTD is the process by which synaptic strength is decreased. It occurs when the presynaptic neuron fires just after the postsynaptic neuron. The further apart these spikes occur, the weaker the synaptic connection becomes. LTD ensures that connections which do not contribute effectively to the postsynaptic activity are weakened. This pruning of less useful synapses is crucial for maintaining the efficiency of the neural network, preventing over-excitation, and ensuring that only the most relevant and effective synaptic pathways are strengthened.

In summary, the STDP learning process can be mathematically simulated using the following equation:

$$\frac{dw_{ij}}{dt} = -A_-(w_{ij})y_i(t)\Sigma_f\delta(t - t_j^f) + A_+(w_{ij})x_j(t)\Sigma_f\delta(t - t_i^f)$$

2.3 Parameter Configurations

2.3.1 Input Patterns

For the input pattern, we used two images. These images were fed into a Poisson encoder, which then provided the encoded inputs to the neuron groups.

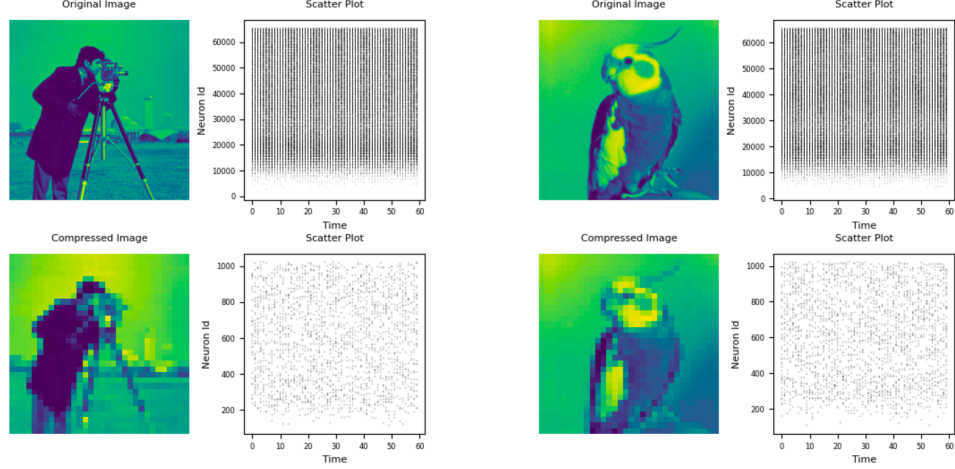


Figure 4: Input images

2.3.2 Spike Trace

The spike trace of each neuron is calculated by adding the current spike and decaying the existing trace. The key parameter in this calculation is the decay rate of the spike trace, which I have set to 1.

2.3.3 STDP Parameters

The most critical parameters in STDP learning are A_+ and A_- , both of which I have set to 3. Additionally, I implemented a hard bound to control the weights.

2.3.4 Neuron Model

in the output layer we need to use a neuron model witch I used LIF neuron model with these parameters:

$$\tau = 10, U_{rest} = -65, U_{reset} = -70, threshold = -40, R = 10$$

3 Lateral Inhibition

Lateral inhibition is the phenomenon in which a neuron's response to a stimulus is inhibited by the excitation of a neighboring neuron. Lateral inhibition has been experimentally observed in the retina and the LGN of organisms. Lateral inhibition makes neurons more sensitive to spatially varying of stimulus than to spatially uniform stimulus. This is because a neuron getting stimulated by a spatially uniform stimulus is also inhibited by its surrounding neurons, thus suppressing its response. On the other hand, a neuron subjected to a spatially varying stimulus is less inhibited by its neighbors that are not excited, thus producing stronger response. Therefore in the case of visual neurons, lateral

inhibition makes them more sensitive to edges on the scene. Although usually described for visual neurons, lateral inhibition is also found in other sensory systems, such as auditory and olfactory neurons. The total region, to which a particular neuron is sensitive to, is called the receptive field of the neuron.

The goal of this experiment is to determine if Lateral Inhibition can enhance the neural differentiation process, leading to a clearer and more distinct pattern recognition capability in the output neurons.

3.1 Different Input Pattern

For this experiment, we employed the first network structure with two distinct input patterns as detailed in the input patterns section. Figure ?? illustrates these patterns. Additionally, we incorporated a lateral inhibition mechanism in the output layer.

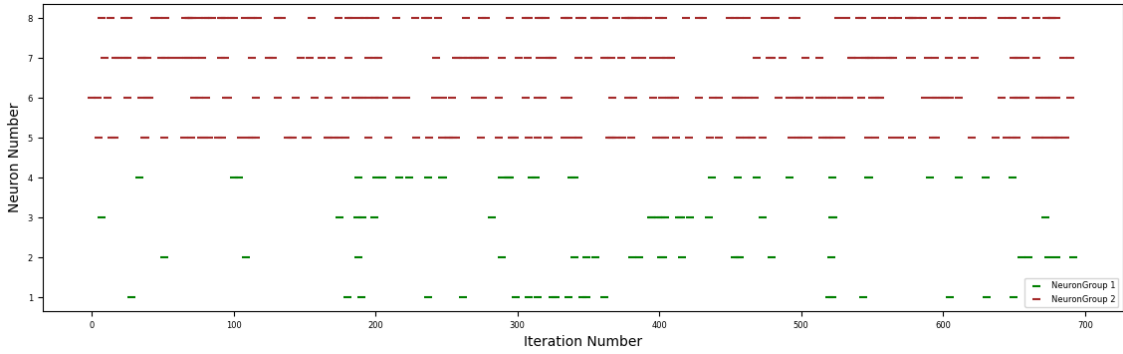


Figure 5: Input Patterns

After 700 iterations of simulation, the resulting changes in weights are visible in Figure 6. Subsequently, we calculated the cosine similarity between the weight vectors connected to the first and second output neurons. Specifically, each of the eight neurons in the input layer is connected to both the first and second output neurons, and we computed the cosine similarity of these weight vectors in every iteration. Figure 21 illustrates the evolution of cosine similarity during the simulation process.

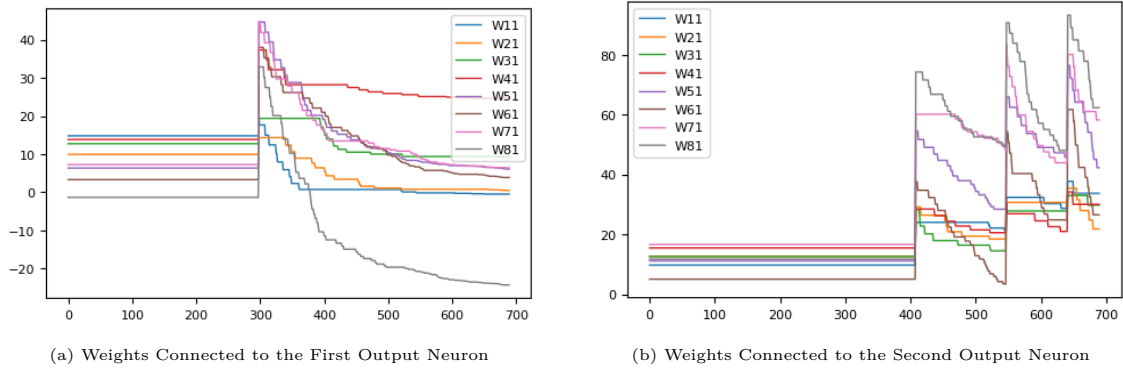


Figure 6: Weight Changes Throughout The Simulation

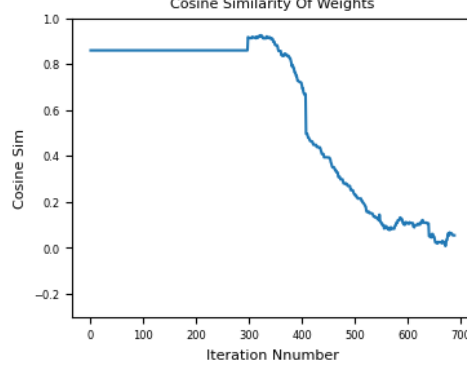


Figure 7: Cosine similarity during the simulation process

If our task is classification, each output neuron represents a distinct class. The model performs effectively when the similarity between these output neurons is minimized, indicating clear differentiation among classes. Throughout the simulation, we observed a decrease in similarity between weights, illustrating the learning of distinct patterns by the output neurons. This outcome is desirable given our use of two distinct input patterns and the implementation of lateral inhibition in the output layer, which aimed to enhance differentiation. Consequently, the application of lateral inhibition increased the distinction between weights.

3.2 Different Input Pattern With Intersection

In this section, we provide the same input pattern to the first and second neurons of both neuron groups in the input layer. Thus, four out of the eight neurons in the input layer share the same pattern. The input patterns are illustrated in Figure 11.

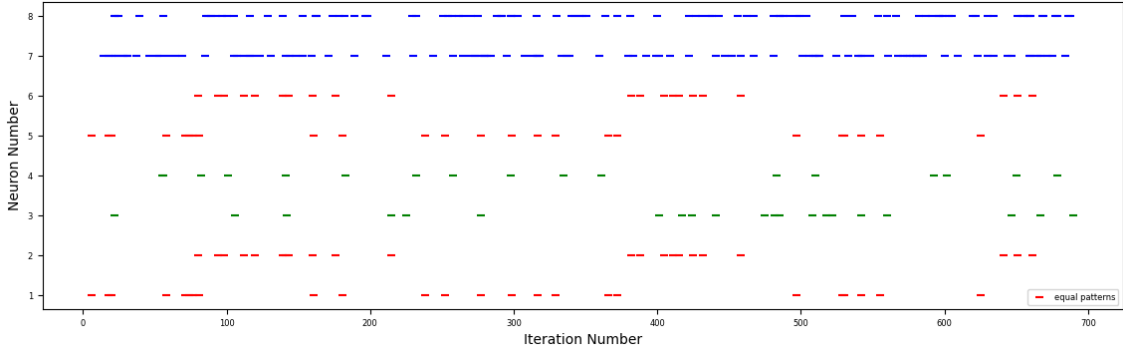


Figure 8: Input Patterns

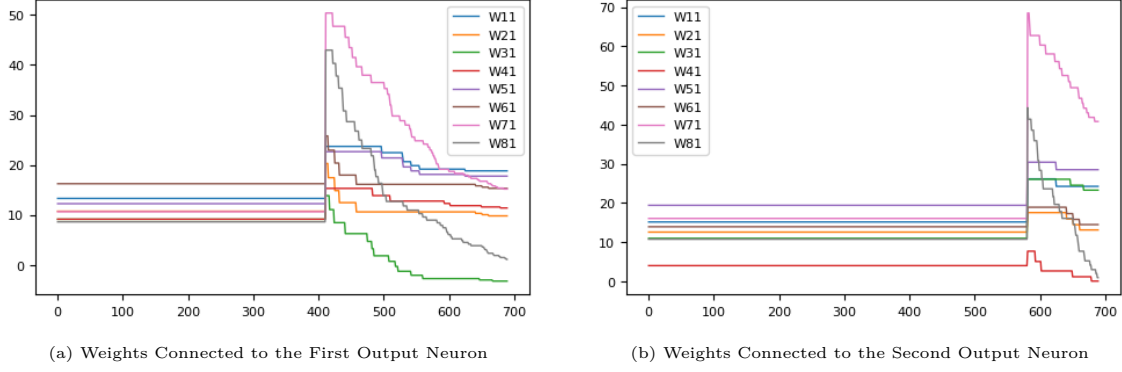


Figure 9: Weight Changes Throughout The Simulation

As seen in Figure 13, throughout the simulation, we observed a decrease in similarity between weights, illustrating the learning of distinct patterns by the output neurons. However, compared to Figure 21, the decrease in weight similarity is much less. This indicates that increasing the overlap of input patterns makes differentiation harder. Nonetheless, with the use of lateral inhibition, there is still a small degree of distinction.

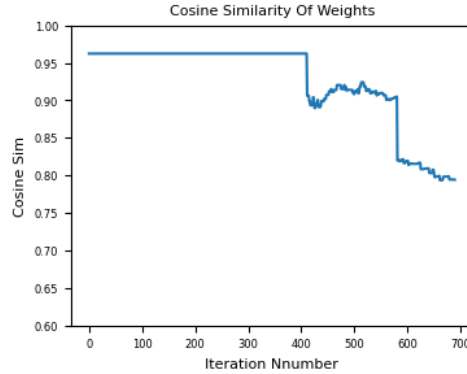


Figure 10: Cosine similarity during the simulation process

3.3 Same Input Pattern

In this section, we provide the same input pattern to our input layer neurons. The input patterns are illustrated in Figure 11.

As expected in this experiment, we provided identical input patterns to both input neuron groups. Initially, with weights following a normal distribution with a standard deviation of 10, the cosine similarity is approximately 1. After several simulation iterations, the cosine similarity remains around 1 because the input patterns are identical. Consequently, even lateral inhibition cannot differentiate between the patterns, which is expected behavior for our network.

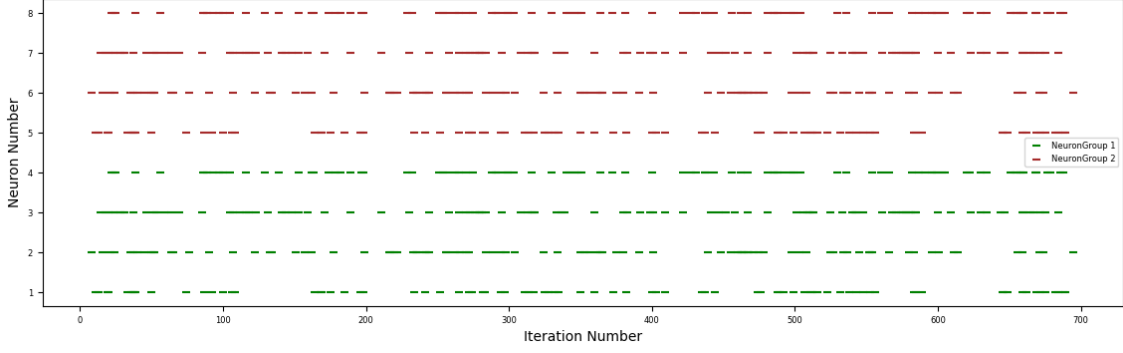


Figure 11: Input Patterns

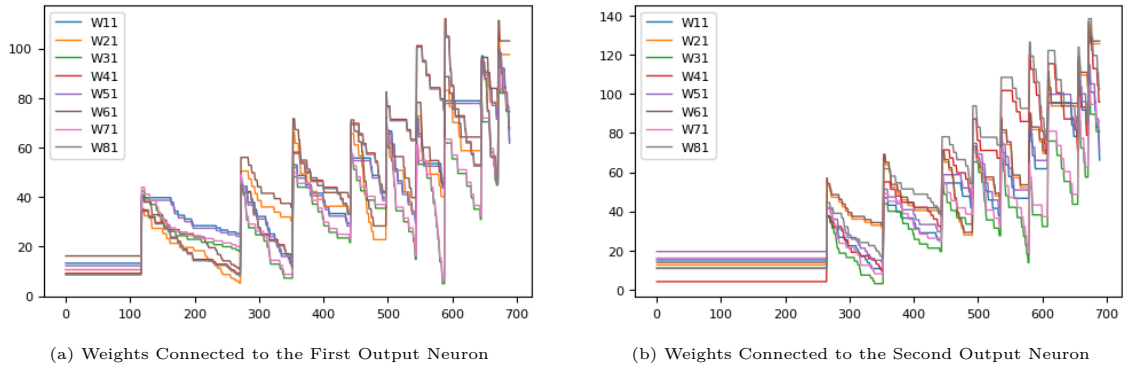


Figure 12: Weight Changes Throughout The Simulation

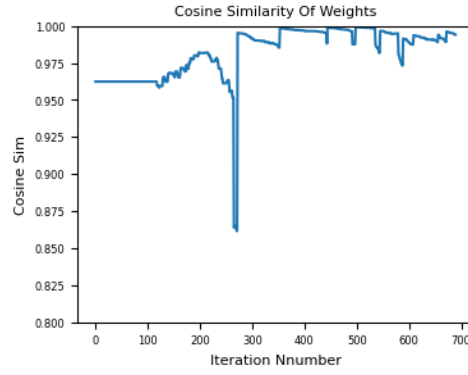


Figure 13: Cosine similarity during the simulation process

4 KWTA¹

K-Winners-Take-All (KWTA) spiking neural networks are designed to select K elements from a total of N elements based on their activity levels, where the K selected elements

¹K Winners Take All

have higher activation values than the remaining $N - K$ elements. When $K = 1$, the KWTA network functions as a Winner-Take-All (WTA) network, which identifies the neuron with the maximum activation.

Selecting the K largest elements from a set of N real numbers is crucial in decision-making, pattern recognition, associative memories, and competitive learning networks. Such tasks are common in classification problems and are essential for developing spiking neural network classifiers, as well as for solving pattern recognition and classification issues.

The implementation concept of KWTA is that if $v \geq \text{threshold}$, then v is set to v_{reset} , and all other spiked neurons are inhibited.

4.1 Different Input Pattern

For this experiment, we used the first network structure with two distinct input patterns, as described in the input patterns section. Figure ?? illustrates these patterns. Additionally, we incorporated a lateral inhibition mechanism and k-Winners-Take-All (KWTA) in the output layer. Since we have two neurons in the output layer, we set $K = 1$. This setting allows us to distinguish between the two output neurons: the more active neuron inhibits the other, thereby differentiating between the patterns.

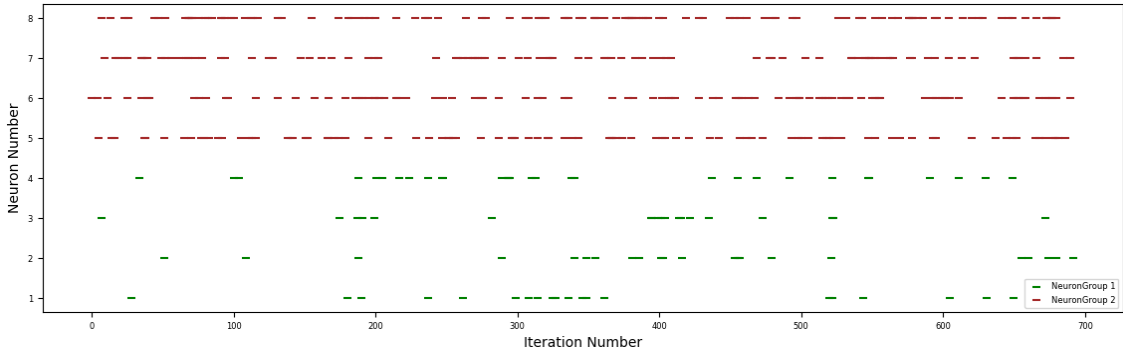


Figure 14: Input Patterns

As shown in Figure 14, the second neuron group exhibits more activity throughout the simulation. Based on the KWTA algorithm, the second output neuron, representing the second neuron group pattern, will inhibit the first output neuron, which corresponds to the first neuron group pattern. Consequently, as depicted in Figure 15, the weights connected to the first output neuron decrease, while the weights connected to the second output neuron increase throughout the simulation.

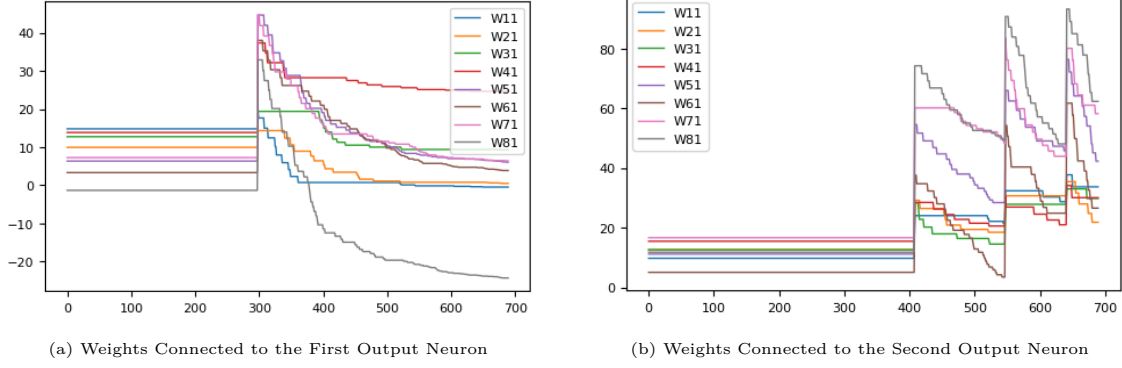


Figure 15: Weight Changes Throughout The Simulation

Finally, Figure 16 shows the cosine similarity between weights throughout the simulation. By adding the KWTa mechanism with $k = 1$ in the output layer, the second output neuron inhibits the first output neuron more than before experiment. Although the effect of KWTa is not very obvious with only two neurons in the output layer, there is a notable decrease in cosine similarity between the weights compared to the before experiment, as shown in Figure 21 and Figure 16. This indicates that the KWTa mechanism has helped to create a greater distinction between the patterns.

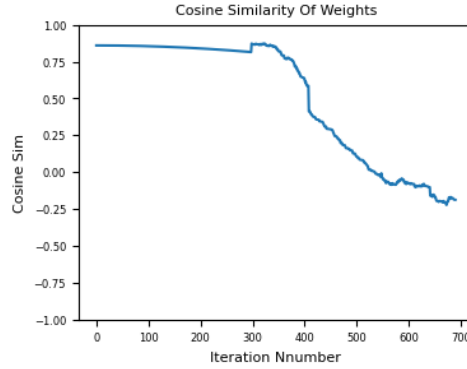


Figure 16: Cosine similarity during the simulation process

4.2 Different Input Pattern With Intersection

In this section, we provide the same input pattern to the first and second neurons of both neuron groups in the input layer. Thus, four out of the eight neurons in the input layer share the same pattern. The input patterns are illustrated in Figure ??.

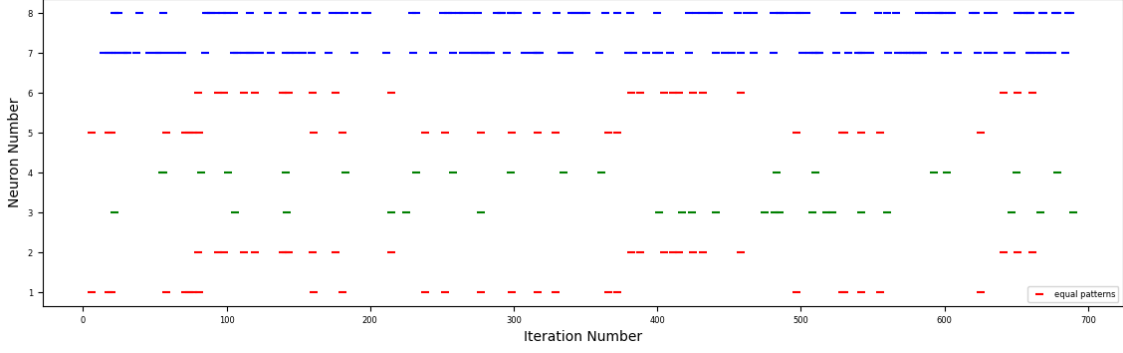


Figure 17: Input Patterns

For this experiment, as mentioned in the previous section, the second neuron group exhibits more activity, inhibiting the first output neuron. As a result, we observe a decrease in the weights connected to the first output neuron and an increase in the weights connected to the second output neuron throughout the simulation. This process is illustrated in Figure 18.

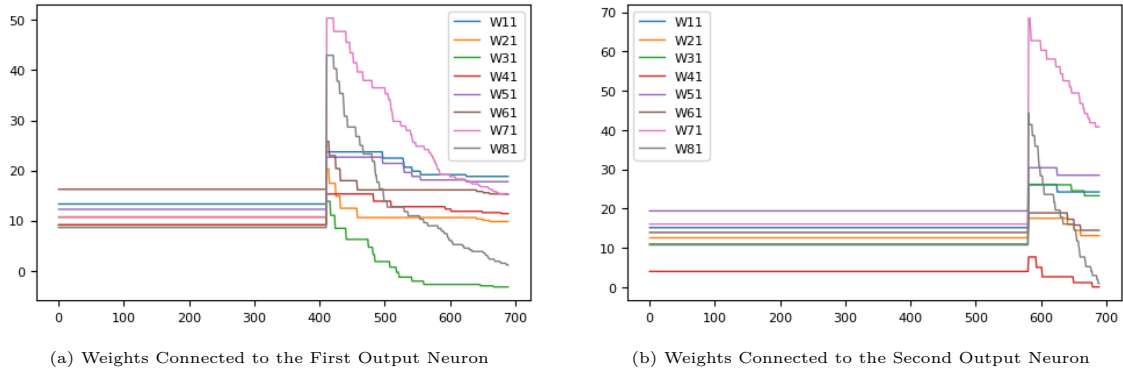


Figure 18: Weight Changes Throughout The Simulation

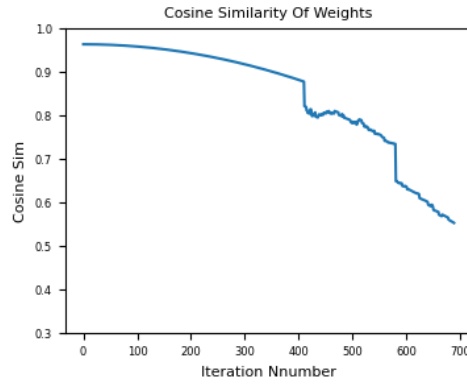


Figure 19: Cosine similarity during the simulation process

In this section, since the patterns have intersections, there is less distinction between them, resulting in a smaller decrease in cosine similarity between weights. Figure 19 illustrates this phenomenon. However, when we used the KWTa mechanism in addition to the lateral inhibition mechanism, we achieved greater distinction between patterns. As shown in Figure 20, KWTa improved the classification of input patterns compared to using only the lateral inhibition mechanism.

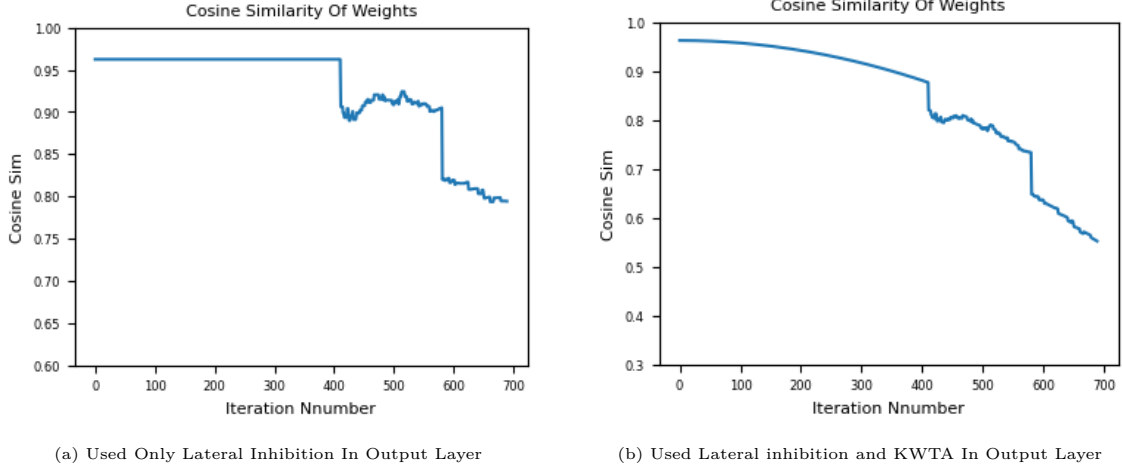


Figure 20: Cosine similarity during the simulation process

4.3 Same Input Pattern

As I mentioned in 3.3, when we have identical input patterns, it is evident that we cannot achieve any distinction. Even with the addition of KWTa, the behavior remains the same. There are minor weight changes due to the randomness of STDP, but throughout the simulation, the weights tend to become identical.

5 Homeostasis

5.1 Homeostatic Process

The human body consists of trillions of cells working together to maintain the organism's overall health and functionality. Despite their diverse functions, these cells share similar metabolic requirements. To ensure the well-being of both individual cells and the entire organism, it is crucial to maintain a constant internal environment that provides necessary elements such as oxygen, glucose, mineral ions, and waste removal. The various processes that regulate this internal environment are collectively known as homeostasis.

5.2 Homeostasis

Homeostasis refers to the stability, balance, or equilibrium of the body's internal environment. It involves the body's continuous efforts to maintain a constant and balanced state through persistent monitoring and adjustments in response to changing conditions. This process, known as homeostatic regulation, comprises three key components: the receptor, the control center, and the effector.

- The Receptor: Detects changes in the environment.
- The Control Center: Receives and processes information from the receptor.
- The Effector: Responds to the control center's commands by either opposing or enhancing the stimulus.

These components work together to restore and maintain homeostasis. For example, in regulating body temperature, temperature receptors in the skin send information to the brain (the control center), which then signals effectors such as blood vessels and sweat glands in the skin. Due to the constant changes in both the internal and external environment, the body must continuously make adjustments to stay at or near a specific value, known as the set point.

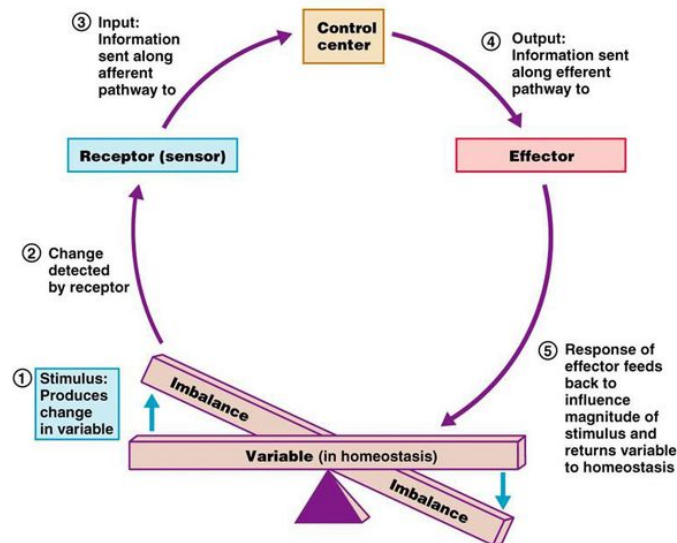


Figure 21: Homeostasis: positive/ negative feedback mechanisms example

5.3 Purpose of Homeostasis

The primary goal of homeostasis is to maintain equilibrium around the set point. While there are normal fluctuations from this set point, the body's systems strive to return to

it. When a stimulus causes a change in the internal or external environment, a receptor detects this change, and the system responds by adjusting the parameter back towards the set point. For instance, if the body becomes too warm, mechanisms are activated to cool it down. Similarly, if blood glucose levels rise after a meal, the body adjusts by moving glucose into tissues that need it or storing it for later use, thereby lowering the blood glucose level.

By continuously regulating these processes, homeostasis ensures the stability necessary for the cells and the organism to function optimally.

5.4 Experiments

In this experiment, we employed the second network architecture detailed in Section 2.1.2. For the input, we utilized five distinct patterns, each representing different images encoded using Poisson coding, as illustrated in Figure 22.

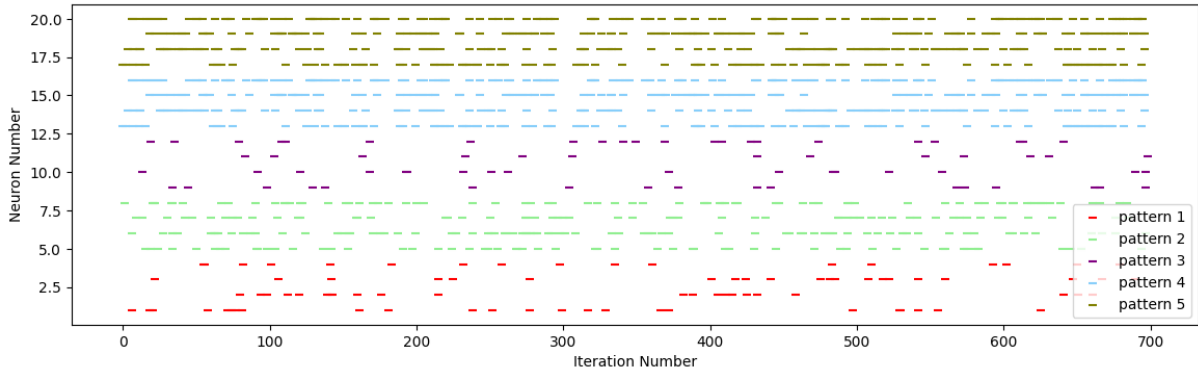


Figure 22: Input Patterns

In our experiment, we utilized five output neurons, with each neuron corresponding to one of the distinct input patterns. Additionally, 20 neurons from the input layer were connected to each of these output neurons, facilitating a complex network of interactions.

Figure 23 illustrates the weight changes for a specific output neuron throughout the simulation process. Given the diversity of input patterns, the weight changes were not uniform across the neurons. However, some patterns exhibited similarities in their weight changes. For instance, the weight changes for the fourth and second neuron groups were quite similar, reflecting the likeness in their input patterns. A similar situation was observed for the first and third neuron groups, where the weight changes closely likely each other. These similarities indicate that input patterns with analogous features result in comparable weight adjustments during the learning process.

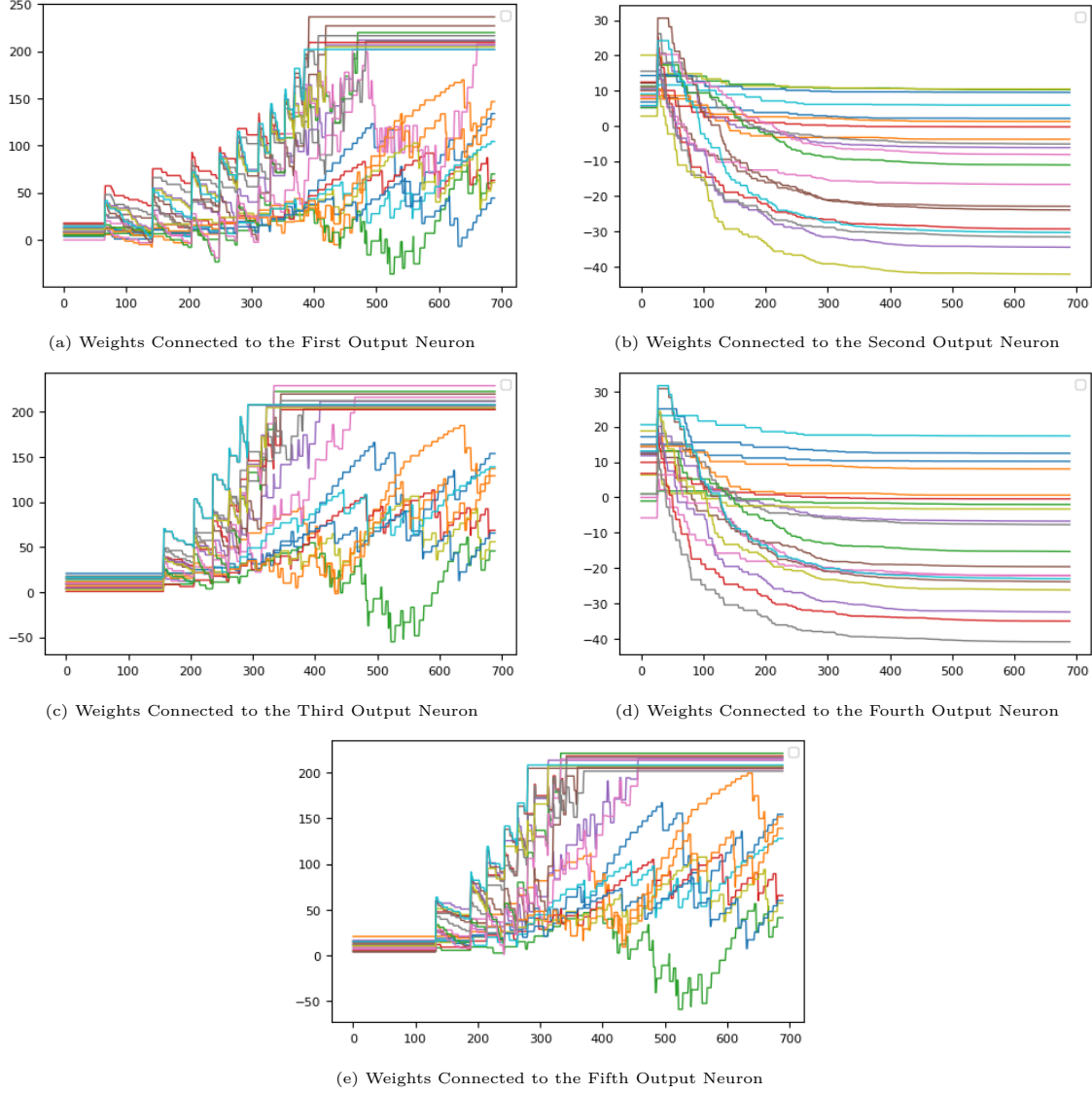


Figure 23: Weight Changes Throughout The Simulation

Finally, for each combination of weights, we defined the cosine similarity of the weights throughout the simulation process. We have 10 different combinations. Figure 24 shows the results of the cosine similarities.

5.5 Conclusion

One of the challenges we encounter in classifying these patterns is the variable distribution of spike counts within the input data. For instance, some activity classes, like the second input pattern, exhibit a large number of active neurons, whereas other classes, like the third input pattern, involve only a few active neurons. This discrepancy in distribution across different classes can introduce a bias in the learning process, favoring high-activity

classes because the neurons corresponding to these classes become the winners. Moreover, the learning progress for classes with lower neuron activity typically proceeds at a slower pace. To mitigate this issue, we employed a decision homeostasis mechanism aimed at addressing these challenges.

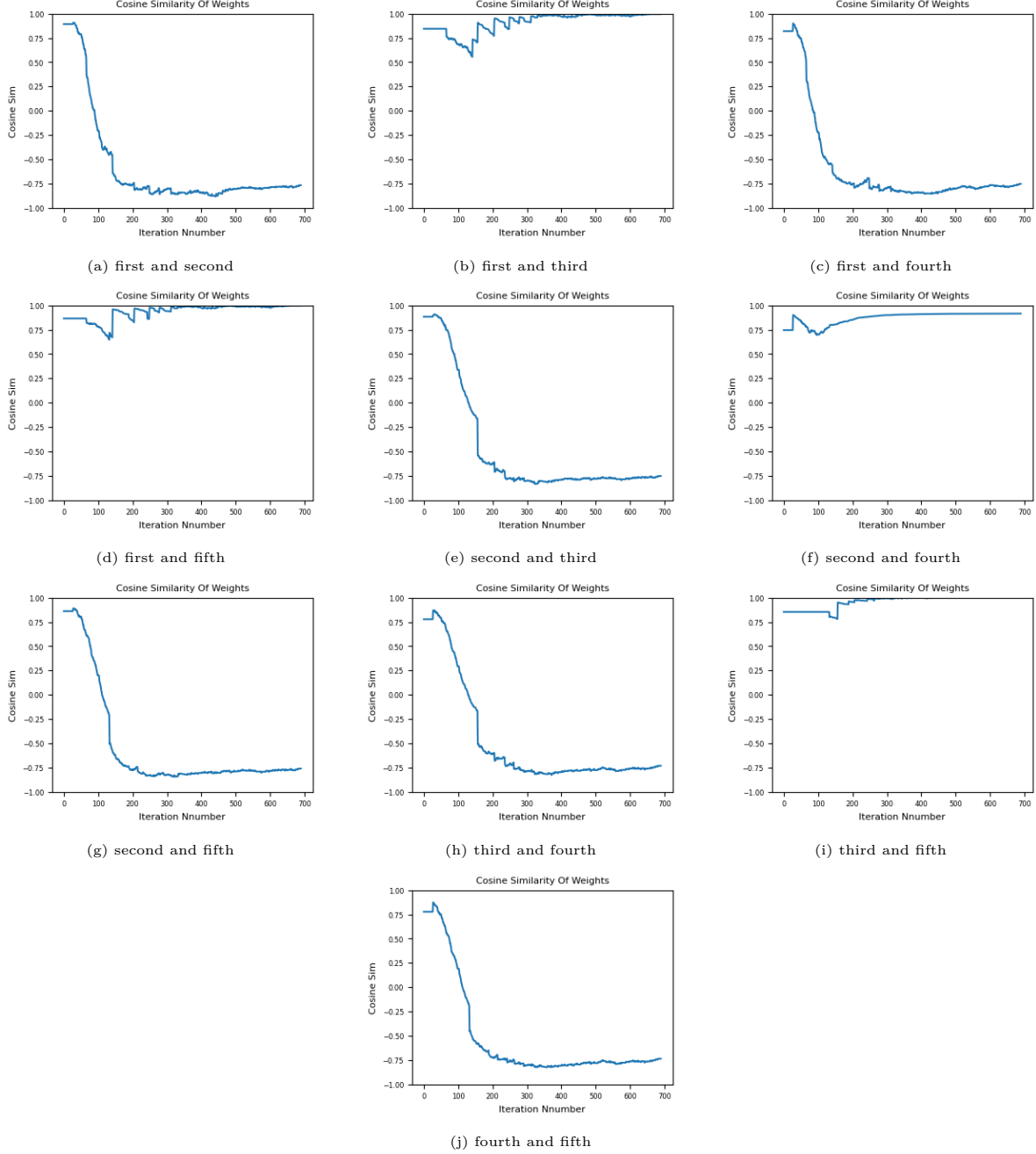


Figure 24: Cosine similarity between the input weights of two output neurons during the simulation process