Image Classification using SNN architecture with different Brain Learning Algorithms

Session Monsoon 2020

Submitted By

Pranika Kakkar

(1710110253)

Rishika Chaudhary

(1710110276)

Under Supervision

of

Dr. Sonal Singhal

(Assoc. Prof., Electrical Engineering Department)

Department Of Electrical Engineering
School Of Engineering
Shiv Nadar University

(Spring 2020)

ABSTRACT

Neuromorphic computing research emulates the neural structure of the human brain. The computational building blocks within neuromorphic computing systems are logically analogous to neurons. A good understanding of the neural processing units and mechanisms and how they are amalgamated to build functioning systems. Therefore, in recent years there is a burgeoning interest in how spiking neural networks can be utilized in order to perform complex computations or solve various tasks like pattern recognition, speech recognition, image classification etc. Training strategy for SNNs can be broadly categorized into unsupervised and supervised algorithms. Unsupervised algorithms discover the characteristics and underlying structures of input patterns without using the corresponding output labels. Spike-Timing-Dependent-Plasticity (STDP) is a bio-plausible unsupervised learning mechanism that instantaneously manipulates the synaptic weights based on the temporal correlations between pre- and postsynaptic spike timings. To delineate former work in this area we will incorporate variations in STDP learning algorithms along with careful analysis of their accuracies. For example, exponential STDP, rectangular STDP, STDP + SVM, etc.In this work a simplified neuron model with a brain-learning algorithm will be used to implement SNN for image classification tasks by using python. Python code has been implemented for the network elements which are neuron, synapse and receptive fields.

TABLE OF CONTENTS

Title **Page No.**

ABSTRACT . . . . **i**

TABLE OF CONTENTS . . . . **ii**

LIST OF TABLES . . . . **iii**

LIST OF FIGURES . . . . **iv**

1 Introduction **1**

1.1 Spiking Neural Network 1

1.1.1 Why SNN? 2

1.1.2 Pre-Post synapse mechanism 2

1.2 Brain Learning Algorithms 4

1.3 Motivation 5

1.4 Problem Statement 6

2 Literature Survey **7**

3 Work Done **9**

3.1 Network Elements 9

3.1.1 Neurons 9

3.1.2 Synapse 11

3.1.3 Receptive Field 11

4 Plan for Next Half Semester **12**

References **13**

List of Tables

3.1 Various Parameters used in the code along with their values 10

4.1 Classification accuracy of spiking neural networks on MNIST test set. . 12

List of Figures

1.1 Basic model of a spiking neuron (Image credit: EPFL) 1
1.2 Postsynaptic potential function PSP with weight dependency -Source(EURASIP Journal on Image and Video Processing) 3
1.3 The STDP learning curve 5

3.1 The workflow and the main components of the spiking neural network . 10
3.2 Spike train fed to neuron 11

Chapter 1

Introduction

Neuromorphic computing has been been around for decades in the field of artificial intelligence. Particularly in the field of image classification, the data is now being encoded into spike trains and seems ideal for benchmarking of neuromorphic learning algorithms [1]. The concept that neural information is encoded in the firing rate of neurons has been the dominant paradigm in neurobiology for many years. This paradigm has also been adopted by the theory of artificial neural networks. Recent physiological experiments demonstrate, however, that in many parts of the nervous system, neural code is founded on the timing of individual action potentials. This finding has given rise to the emergence of a new class of neural models, called spiking neural networks.

1.1 Spiking Neural Network

The spiking neural networks (SNNs) are artificial neural networks that imitate natural neural networks. They include the concept of time along with neuronal and synaptic state. The neurons in the SNN do not fire at every propagation cycle, instead fire only when the membrane potential reaches a specific threshold value. When a neuron is fired it generate signals that travel to other neurons which, in turn change their potentials proportionally with the signal (it can either increase or decrease).

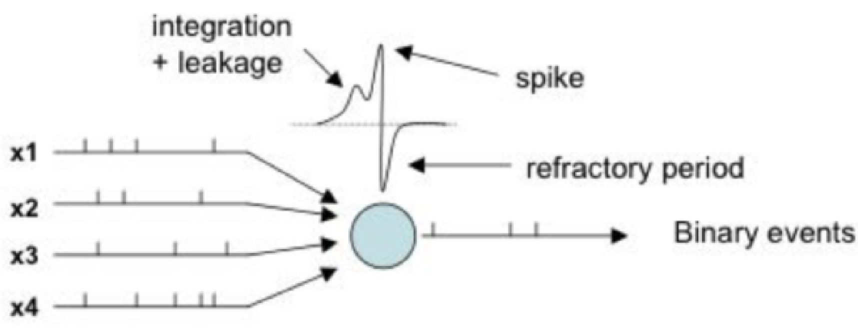

Figure 1.1: Basic model of a spiking neuron (Image credit: EPFL)

The SNN model used in this work is the feed-forward network, each neuron is connected to all the neurons in the next layer by a weighted connection, which means that the output signal of a neuron has a different weighted potential contribution . Input neurons require spike trains and input signals (stimuli) need to be encoded into spikes (typically, spike trains) to further feed the SNN.
The figure 1.1 delineates the basic model of a spiking neural network where x1, x2, x3, x4 are the inputs for the model. After several incoming spikes, the membrane potential surpasses threshold and neuron fires a postsynaptic spike.

1.1.1 Why SNN?

- Spiking Neural Network's overcome the computational power of neural networks made of threshold. They open up new perspectives for developing models with an exponential capacity of memorizing and a strong ability to fast adaptation.Moreover, SNNs add a new dimension, the temporal axis, to the representation capacity and the processing abilities of neural networks[3].

- The principal difference between a traditional ANN and SNN is the neuron model that is used. In a traditional ANN, it does not engage individual spikes in computations. Instead the output signals generated from the neurons are treated as normalised firing rates. This is an averaging procedure and it is called rate coding. Instead of using rate coding, SNN uses pulse coding. Each individual spike is used in the neuron model of an SNN. The main characteristic here is incorporation of timing of the firing in computations, like real neurons do.

- In traditional ANN, one of the most common learning algorithm is the Stochastic Gradient Descent. To calculate the gradient of the loss function with respect to the weights, most state of the art ANNs use a procedure called back-propagation. However, the biological plausibility of back-propagation remains highly debatable. For example, there is no evidence of a global error minimisation mechanism biological neurons[3] .

1.1.2 Pre-Post synapse mechanism

One of the major features of a spiking neuron is the membrane potential, the transmission of an individual spike from one neuron to another is mediated by synapses. A pre-synaptic neuron is a transmitting neuron and a postsynaptic neuron as a receiving neuron. In inactive stage the neurons possess a small negative electrical charge of -70 mV, known as the resting potential. When a single spike arrives into a postsynaptic neuron, it generates a post synaptic potential which is excitatory when the membrane potential is increasing and inhibitory when decreasing. The membrane potential at an instant is the sum of all present PSP at the neuron inputs. The membrane potential generates a postsynaptic spike after it overcomes a critical threshold value. It further enters the neuron into a refractory period when the membrane remains in an over polarized state consequently preventing generation of new spikes by the neuron temporarily. Post the refractory period, the neuron potential goes back to its resting value and when membrane potential is above the threshold, it fires a new spike.

This figure 1.2 describes different PSP as a function of time and weight value, (a) red line and the blue line is for excitatory, and the green line is inhibitory . (b) Two neurons (yellow) generate spikes, which are presynaptic for next layer neuron (green).Part (c) shows the membrane potential graph for green neuron. Presynaptic spikes raise the potential; when the potential is above threshold, a postsynaptic spike is generated and the neuron becomes over polarized [2].

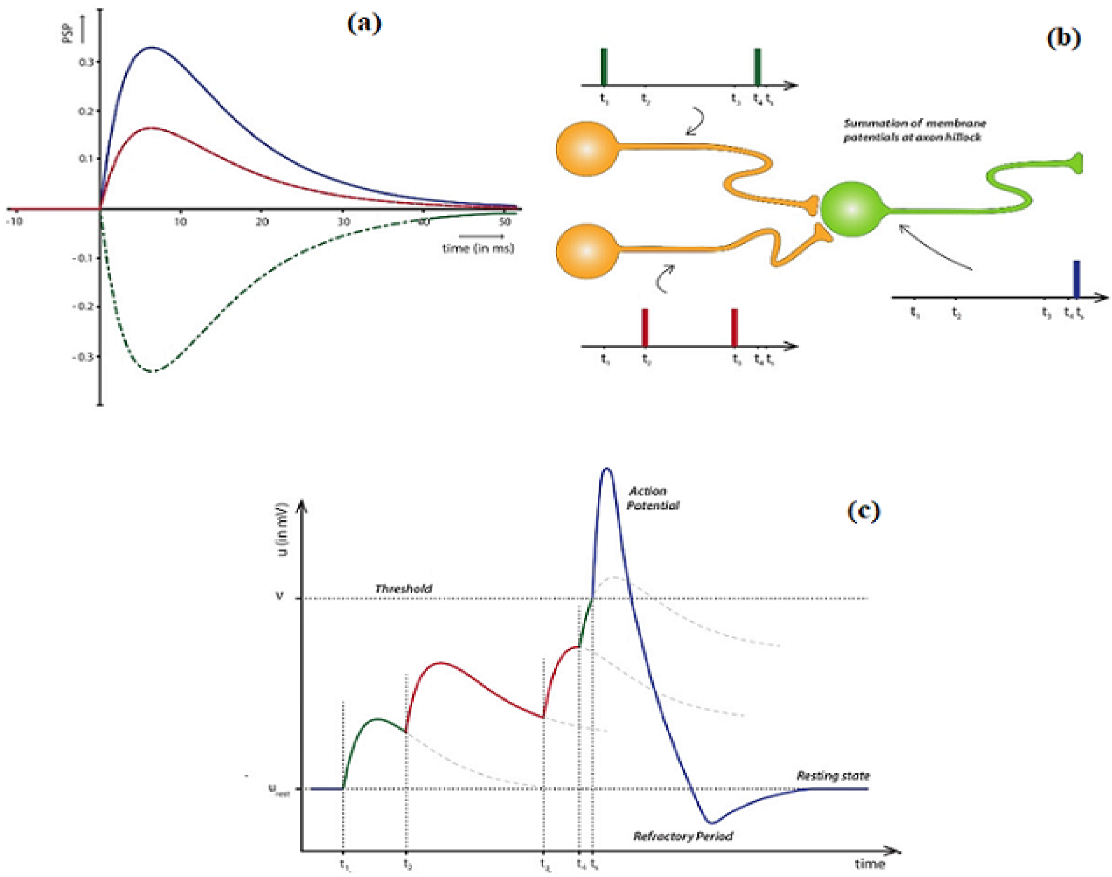

Figure 1.2: Postsynaptic potential function PSP with weight dependency - Source(EURASIP Journal on Image and Video Processing)

1.2 Brain Learning Algorithms

The changing and shaping of neuron connections in our brain is known as synaptic plasticity. Neurons fire, or spike, to signal the presence of the feature that they are tuned for. When two neurons fire at almost the same time the connections between them are strengthened and thus they become more likely to fire again in the future. When two neurons fire in an uncoordinated manner the connections between them weaken and they are more likely to act independently in the future. This is known as Hebbian learning. The strengthening of synapses is known as Long Term Potentiation (LTP) and the weakening of synaptic strength is known as Long Term Depression (LTD). What determines whether a synapse will undergo LTP or LTD is the timing between the pre- and postsynaptic firing. If the presynaptic neuron fires before the postsynaptic neuron within the preceding 20ms, LTP occurs; and if the presynaptic neuron fires after the postsynaptic neuron within the following 20ms, LTD occurs. This is known as Spike-Timing Dependent Plasticity (STDP) [3].

STDP is an unsupervised learning method. This is also a reason why STDP-based learning is believed to more accurately reflect human learning, given that much of the most important learning we do is experiential and unsupervised, i.e. there is no "right answer" available for the brain to learn from.

There are also a lot of variants of STDP which are undergoing extensive research.It is sometimes cumbersome to maintain the functionality of neural circuits unless, the changes in synaptic strength are coordinated across multiple synapses along with other neuronal properties. Homeostatic plasticity is one of these that includes schemes that control the total synaptic strength of a neuron, that regulate its intrinsic excitability as a function of average activity, or that make the ability of synapses to undergo Hebbian leaning depend upon their history of use. Another variant is STDP + SVM.Unsupervised learning with STDP has been demonstrated with both rate based poisson coding and latency based one spike per neuron coding. Convolution and pooling layers in cascade with the network performing layer by layer learning to learn hierarchical features has shown the best performance reported to date for an unsupervised learning SNN[4]. The Backpropagation using STDP model learning rules are inspired from the backpropagation update rules reported for neural networks that are equipped with ReLU activation function[5].

The following is the learning curve for STDP, here the curve has strong depression value than potentiation, increasing specificity[2].

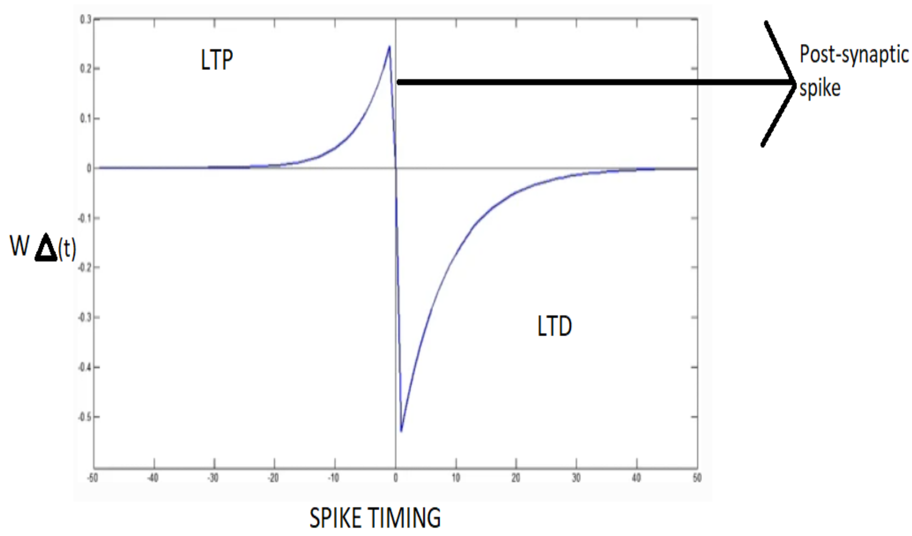

Figure 1.3: The STDP learning curve

The following are the equations used in the weight updation rule for STDP learning algorithm:

$$W\Delta(t)=\begin{cases} A_+e^{-\Delta/\tau_+}, & if\Delta t \geq 0 \\ -A_-e^{\Delta/\tau_-}, & if\Delta t \ < 0 \end{cases}$$

$$\mathrm{W}_{new}=\begin{cases} w_{old}+\sigma\Delta w(w_{max}-w_{old}), & if\Delta w > 0 \\ w_{old}+\sigma\Delta w(w_{old}-w_{min}), & if\Delta w \ \leq 0 \end{cases}$$

1.3 Motivation

Spiking Neural Networks (SNNs) show a lot of potential as a candidate for human brain motivated neuromorphic evaluation because of their innate power efficiency and noteworthy deduction accuracy for many cognitive problems like image classification and speech recognition. The human brain has robust unsupervised learning ability that is it can function without a data samples that have already been labelled. Therefore, to corroborate brain-inspired neural networks have gained profound interest. The signals is transmitted as spike sequences whereas the conventional neural networks consists of numerical data. Thus, it is not astounding that that the at present most famous models in machine learning, artificial neural networks (ANN) or deep neural networks are enlivened by highlights found in biology.The thought of STDP has been demonstrated to be a verified learning algorithm for forward-connected artificial neural networks in numerous applications. To name a few-pattern recognition, Recognising traffic, sound or movement using Dynamic Vision Sensor (DVS) cameras, gender recognition are the related areas that have been already explored.

1.4 Problem Statement

- To implement the SNN architecture for embedded application such as image classification.
- Implementation of different existing brain learning algorithm on the SNN model.
- Analysing and comparing different brain learning algorithms and storing their respective accuracies.

Chapter 2

Literature Survey

M. M. Taylor in the year 1973 had suggested that when Hebbian learning occurred that is related to strengthening of synapses for which a presynaptic spike occurred just before a postsynaptic spike, while in anti-Hebbian learning synapses weakened, without the absence of a closely timed presynaptic. Currently, one such research project is the Human Brain Project which brings together neuroscientists, computer and robotics experts to build a unique Information and Communications Technology (ICT)-based infrastructure for brain research. The Human Brain Project is in association with ongoing initiatives in Europe and beyond USA, Canada and Japan. They aim to boost the understanding of brain through modelling and simulation in computers.

Peter U. Diehl et.al [6], presented a SNN which relies on a combination of biologically plausible mechanisms and uses unsupervised learning.Dataset considered for training and testing for the proposed SNN is MNIST dataset. The performance of the approach reported as 95 % using 6400 learning neurons. They have observed that later inhibition has generated competion among neurons and homoeostasis helps in giving each neuron an equal chance to compete. The input to the network (MNIST) dataset, it contains 60,000 training examples and 10,000 test examples with the input images having pixel size of 28x28 of digits 0-9. The input is in form of Poisson spike trains having the firing rates proportional to intesity of pixel images. Those Poisson-spike trains are fed as input to excitatory neurons. These neurons are connected to inhibitory neurons via one-to-one connections. This connectivity provides lateral inhibition and leads to competition among excitatory neurons. The presented network on the MNIST dataset achieved good classification performance using SNNs with unsupervised learning made of biologically plausible components.

Reference [5], proposes a temporally local learning rule that uses the backpropagation weight change updates applied at every time step. The aforementioned techique has two folds benefits an accurate gradient- descent and temporally local, efficient STDP. The above approach is experimented on three different data sets, namely the XOR problem, the Iris data, and the MNIST dataset. The following results were obtained in the paper -STDP-based backpropagation 3-layer SNN; H1=500; H2=150, where H1 and H2 are the number of neurons in the hidden layers with an accuracy of 97.20%. Therefore, a network of spiking IF neurons can undergo backpropagation learning applied to conventional NNs demonstrating along with the accuracy obtained with the datasets show that SNN performs as successfully as the traditional NNs. SNN receives spike trains representing input feature values in T ms. The learning rules and the network status in the SNN are specified by an additional term as time (t)The major difference between the shown two networks is their data communication where the neural network (left) receives and generates real numbers while the SNN (right) receives and generates spike trains in T ms time intervals.[5].

Reference [2], a novel, simplified and computationally efficient model of spike response model (SRM) neuron with spike-time dependent plasticity (STDP) learning is presented. The data representation is done by frequency spike coding based on receptive fields analogous to visual cortex the images are processed and encoded by the network. The network output proves its utility as primary feature extractor for further refined recognition or as a simple object classifier. The results display that the model can successfully learn and classify black and white images with added noise or partially obscured samples with up to 20 computing speed-up at an equivalent classification ratio when compared to classic SRM neuron membrane models. The proposed solution in the paper is an amalgamation of spike encoding, network topology, neuron membrane model and STDP learning.

Chapter 3

Work Done

This chapter describes the building the network elements used in making the Spiking Neural Network. There are 4 network elements. The first is neuron, which is generated by feeding it a random spike train and the membrane potential is updated using the equations given below. The second is synapse, it is the weighted path the created spikes from one neuron to the other linked neurons.A simple network layer is implemented that consists of 2 layers- the first contains 5 neurons and the second layer is made up of 3 neurons. Synapse connects all the neurons in the first layer to every neuron in the second layer. The third is receptive field, in which we make a sliding window 5x5, an on-receptive field was used. This receptive field was weighted according to Manhattan distance. A 16x16 input is processed and encoded.

3.1 Network Elements

The python implementation of hardware efficient spiking neural network .The aim is to build a network which could be used for prediction and on chip-learning.The network elements consists of -

- Neurons
- Synapse
- Receptive Field
- Spike Train

The figure 3.1 delineates the work flow of the implemented code which together constitute the spiking neural network.

3.1.1 Neurons

Neuron is the fundamental unit of an SNN. The input, hidden and output layers consist of several interconnected neurons and this neuron emulates the classic leaky integrate-and-fire (LIF) model [7]. Defining membrane potential P_t as a function of incoming

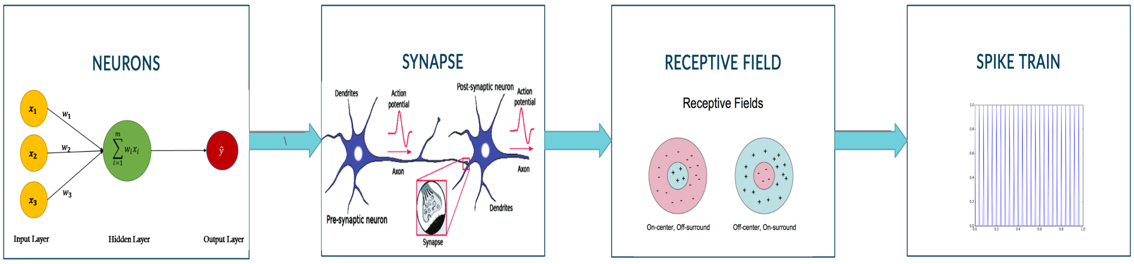

Figure 3.1: The workflow and the main components of the spiking neural network

spikes and times. Since the model is developed to be utilised in digital circuits, the time units are taken in discrete time form. For a n-input SNN taking the time period as non-refractory,every individual incoming input spike S_{it}, i=[1..n] burgeons the membrane potential Pt with respect to the value of the synapse weight W_i. Additionally, the membrane potential decreases by a constant value D, for every time instant. The process is defined by the equation given below−

$$\mathrm{P}_t = \begin{cases} P_{t-1} \text{ - D + } \sum_{n=1}^{n} W_i \, \mathrm{S}_{it} \ , & if P_{min} < \mathrm{P}_{t-1} < \mathrm{P}_{threshold} \\ P_{refract}, & if P_{min} \geq P_{threshold} \\ R_p \ , & if P_{t-1} \leq P_{min} \end{cases}$$

When the potential crosses a threshold value, neuron enters into refractory period in which no new input is allowed and the potential remains constant. To avoid strong negative polarization of membrane, its potential is limited by Pmin. As long as $P(n) > P(\mathrm{min})$, there is a constant leakage of potential.

The neuron implementation was done in python, a random spike train was given as input to neuron and after defining the parameters of the above equation, the same loop was implemented.

Parameters	Values
P_{ref}	0
P_{min}	-1
P_{th}	25
D	0.25

Table 3.1: Various Parameters used in the code along with their values

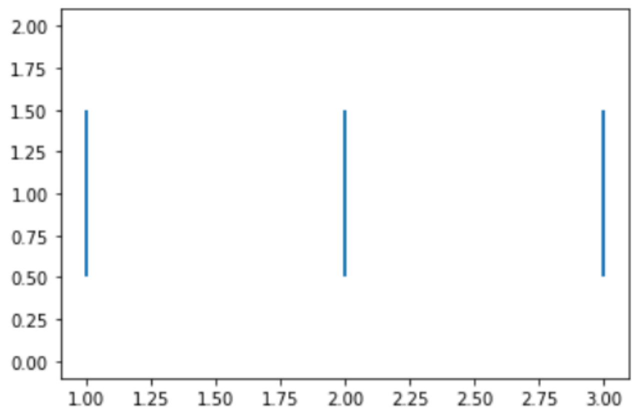

Figure 3.2: Spike train fed to neuron

3.1.2 Synapse

In neurobiology synapse is a junction between two nerve cells, consisting of a minute gap across which impulses pass by diffusion of a neurotransmitter. In an SNN, synapse is the weighted path for generated spikes from one neuron to the other connected neurons. This is the implementation of a simple network of 2 layers with 5 neurons in the first layer and 3 in the second as shown in the figure. Each neuron in the first layer is connected to all the neurons in the second layer via synapse. Synapses are realised by a 2D matrix of size (5x3) initialised with random weights. It can be expanded to any number of layers with any number of neurons in it. Here, we have implemented a two layer network. The first layer contains 5 neurons and the second layer contains 3 neurons. The output of the first layer was calculated and fed into the input of the second layer, and then the output of the second layer was calculated. We get our syapse as a 3x5 matrix.

3.1.3 Receptive Field

Stimulation leads to response of a particular sensory neuron in an area which is known as the receptive field. For spiking neural network, where the image is the input, the receptive field of a sensory neuron is the part of the image which increases its membrane potential. In our implementation, we have used on-centred field. The on-centered receptive field which we have used is 5x5. This receptive field was weighted according to the Manhattan Distance to the center of the field. A 16x16 pixel input is processed by a 16x16 encoding neuron layer(256 neurons), obtaining a potential value for each input which will further be converted into spikes[2].In python, after calculating the Manhattan distance, the membrane potential for each of the 256 neurons was calculated and the output obtained was a 16x16 matrix.

Chapter 4

Plan for Next Half Semester

- To implement our spiking neural model we will use a subset of the Semeion and MNIST set of data, and the above network elements will be combined in the main python file to completely form our network model.
- We also plan to implement various variants of STDP learning and then compare and store their accuracies. The table 4.1 shows the different learning algorithms whose literature will be studied in the next half of the semester.

Architecture	Training Type	Learning-rule
Two layer network	Spike-based(Unsupervised)	Rectangular STDP
Two layer network	Spike-based(Unsupervised)	Exponential STDP
SDNN[4]	Spike-based(Unsupervised)	STDP + SVM
BP-STDP[5]	Spike-based(supervised)	STDP + BP

Table 4.1: Classification accuracy of spiking neural networks on MNIST test set.

- After comparing their accuracies, we will perform hardware implementation on the best algorithm..

Bibliography

[1] R. Gopalakrishnan, Y. Chua, and L. R. Iyer. Classifying neuromorphic data using a deep learning framework for image classification. In *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1520–1524, 2018.

[2] Juan F Guerrero-Martínez Manuel Bataller-Mompeán Taras Iakymchuk, Alfredo Rosado-Muñoz and Jose V Francés-Víllora. Simplified spiking neural network architecture and stdp learning algorithm applied to image classification. *EURASIP Journal on Image and Video Processing*, 2015.

[3] Yi-Ling Hwong. Unsupervised learning with spike-timing dependent plasticity.

[4] A. Vasudevan, T. Serrano-Gotarredona, and B. Linares-Barranco. Learning weights with stdp to build prototype images for classification. In *2019 14th International Conference on Design Technology of Integrated Systems In Nanoscale Era (DTIS)*, pages 1–5, 2019.

[5] Amirhossein Tavanaei and Anthony Maida. Bp-stdp: Approximating backpropagation using spike timing dependent plasticity. *Neurocomputing*, 330:39 – 47, 2019.

[6] Peter Diehl and Matthew Cook. Unsupervised learning of digit recognition using spike-timing-dependent plasticity. *Frontiers in Computational Neuroscience*, 9:99, 2015.

[7] Wulfram Gerstner and Werner M. Kistler. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002.