University of Tehran

COMPUTER SCIENCE DEPARTMENT

COMPUTATIONAL NEUROSCIENCE

Final Project

Image Classification with Homeostasis using Spiking Neural Network

Student Name	Student ID
AMIR NADERI	610398126

Lecturer in charge:

M.Ganjtabesh

Submission Date : 14/08/2023

1 Introduction

In this project, we are going to extract features from images using convolution and max-pooling layer. The model we used has three layers. there is convolutional Connection between first layer and second layer. Also the weights of filters are training by the STDP learning rule. In the following we have pooling connection between second layer and the third layer which the spikes that happened sooner in the last layer will spike here first. Here we have 5 filters with random weigh initialization, these weights will learn the features. The images are in gray scale, so they have just one channel. The size of the kernel of filters are 5.

2 Convolution and Max-Pooling

In mathematics, convolution is an operation performed on two functions (f and g) to produce a third function that expresses how the shape of one is modified by the other. Convolution is one of the most important operations in signal and image processing. It could operate in 1D (e.g. speech processing), 2D (e.g. image processing). In image processing, convolution is the process of transforming an image by applying a kernel over each pixel and its local neighbors across the entire image. The kernel is a matrix of values whose size and values determine the transformation effect of the convolution process. The Convolution Process involves placing the Kernel Matrix over each pixel of the image, multiplies each value of the Kernel with the corresponding pixel it is over, Then sums the resulting multiplied values and returns the resulting value as the new value of the center pixel. This process is repeated across the entire image. So if we were to process the image of higher resolutions the network parameters will be very high and require higher computational power and won't scale for larger images. No one wants to train millions of parameters for the small images, therefore we use convolution layers.

Pooling is the process of extracting the features from the image output of a convolution layer. This will also follow the same process of sliding over the image with a specified pool size and kernel size. Pooling layers reduce the dimensions of data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. Local pooling combines small clusters, tiling sizes such as 2×2 are commonly used. Global pooling acts on all the neurons of the feature map. Max Pooling is being used widely and it will just keep the highest number in the pool and discard the rest. By getting the highest value in each pool we will be getting the significant features of the image, the lower values are not the features at all or not significant features to be able to use in the model.

3 Model

Neurons in this model are LIF. Let's see the parameters of the LIF model:

- τ: 10
- R: 5
- θ: -37
- u_{rest}: -67

As we said we have three layers in the model with convolutional and pooling connections. The weights will be learned by the STDP rule. Spike-timing-dependent plasticity (STDP) is a biological process that adjusts the strength of connections between neurons in the brain or a computational model. The process adjusts the connection strengths based on the relative timing of a particular neuron's output and input action potentials (or spikes). Here in this project we used ConvSTDP which you can see in the code. In ConvSTDP the STDP rule applies on the weifgts of the filters. So the filters will learn the frequent features.
See a diagram of the model:

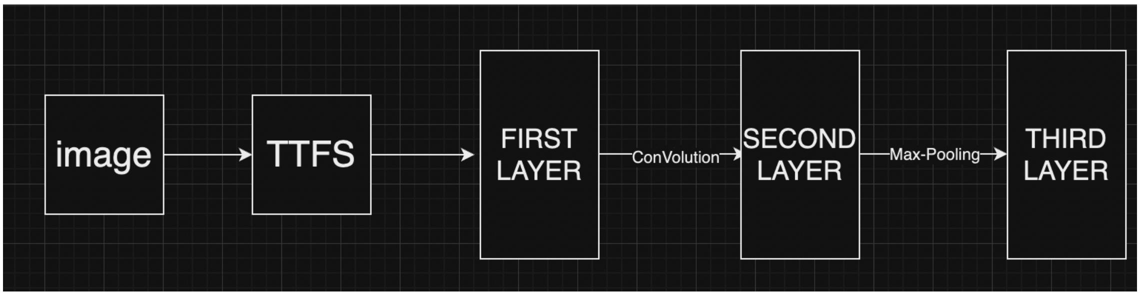

Figure 1: Diagram of the model

So we have three neuron groups with two synapse groups. In the first neuron group there are 28*28 neurons which is the size of our image. Size of the second and third layers are smaller due to the formula of the convolution and max-pooling.

4 Learning features with STDP

First we are using the images of faces and converting them to the gray scale ones. See an example of image:

Figure 2: Picture of a face

Then we apply TTFS encoding on the image. See the result:

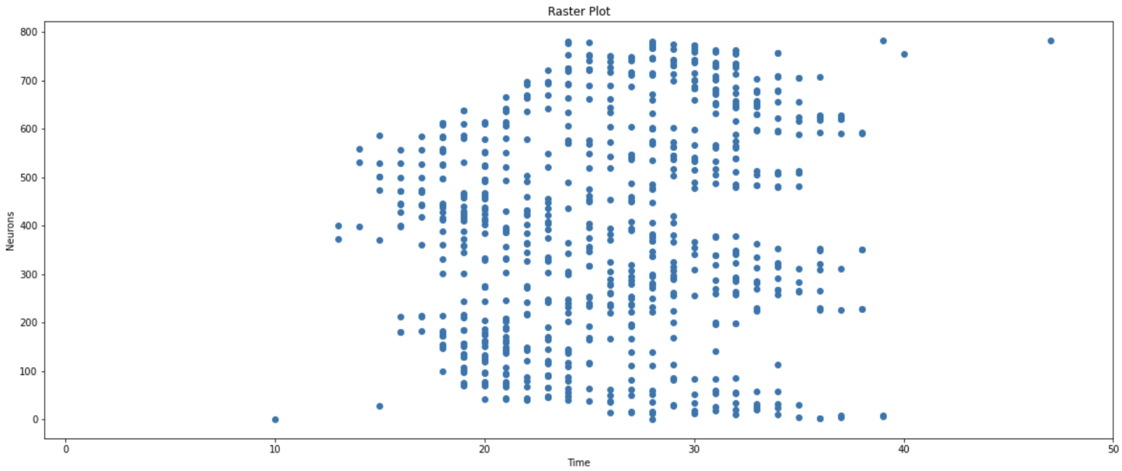

Figure 3: TTFS Encoding

In a pure version of this coding scheme, each neuron needs to fire only a single spike to transmit information. If it emits several spikes, only the first spike after the reference signal counts. All following spikes would be irrelevant. To implement a clean version of such a coding scheme, we imagine that each neuron is shut off by inhibition as soon as it has fired a spike. Inhibition ends with the onset of the next stimulus (e.g., after the next saccade). After the release from inhibition the neuron is ready to emit its next spike, which now transmits information about the new stimulus. Since each neuron in such a scenario transmits exactly one spike per stimulus, it is clear that only the timing conveys information and not the number of spikes. Experimental evidence indicates that a coding scheme based on the latency of the first spike transmit a large amount of information.
You can see the Convolutional connection and Pooling connection in the code. The pattern of images enters the model and after some convolution and max-pooling, by the rule of STDP, our weights got trained. Hence we have activity (spikes) in the layers. Let's see the raster plot of the simulation after training:

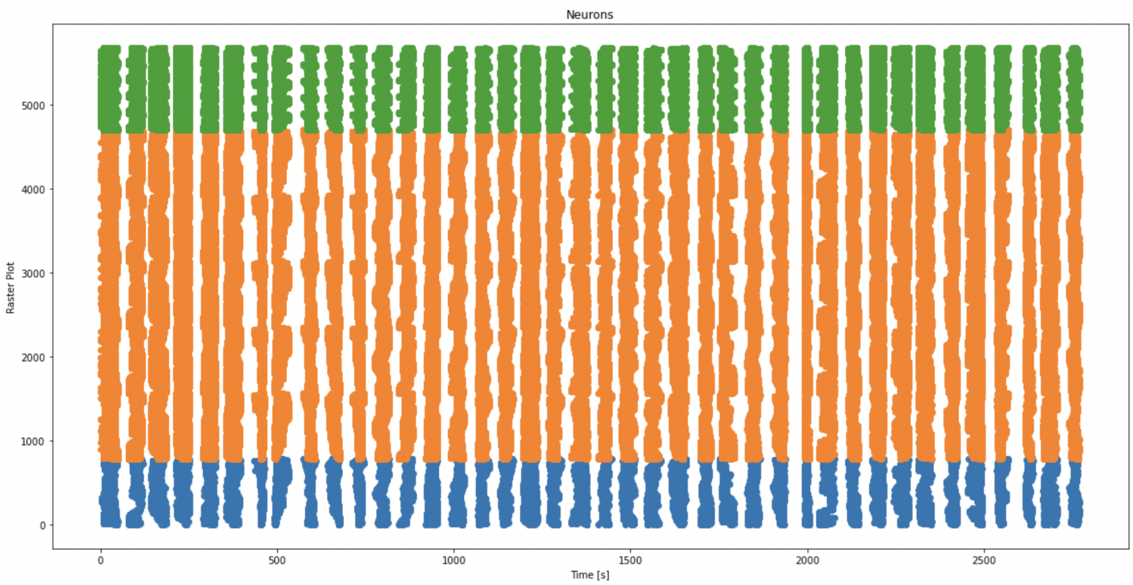

Figure 4: Raster Plot

As you see the blue dots represent the spikes in the first layer which are the TTFS encoding of the images. The orange dots are the activity of neurons in the second layer after convolving our five filters on the spike trains of the last layer. The point is padding is the same here. At last the green dots indicate the spikes in the last (third) layer which is the result of max-pooling. The point is when we are engaging with spiking neural networks, Pooling will have different meaning. Here we are performing local max operation by simply propagating the first spike emitted by a given group of pre-synaptic in the last layer.
We are using STDP, so the frequent features will be learned. The resolution of the images are low due to the lack of computation, therefore the results maybe a little unclear. Now let's see the matrices of weights:

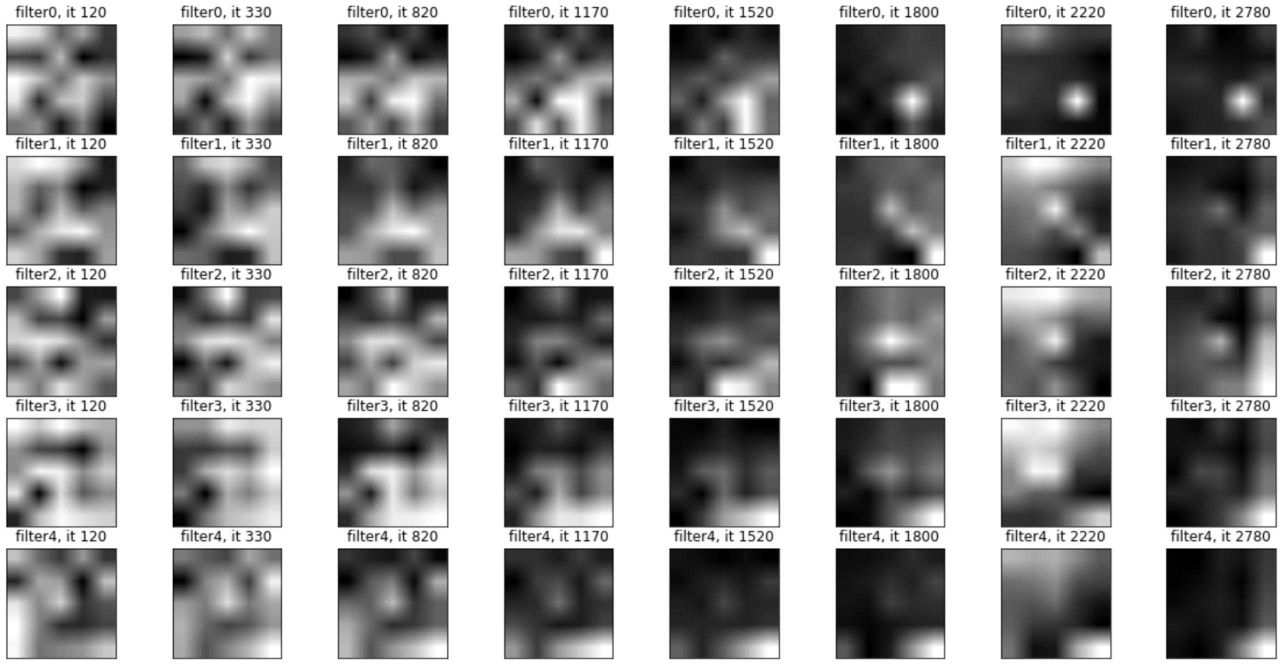

Figure 5: Matrices of Weights with padding same

As you see we have five filters with random initialization. After learning features with STDP, the matrices converged to something that can help us due to extracting features and classify images.
Now if we change any parameters of convolution, we get another features. For example we set the padding to zero, now let's see the matrices of weights:

Figure 6: Matrices of Weights with padding zero

Now we use the dog dataset to see the activity and features learned by this model. First let's see the TTFS encoding of one of the images:

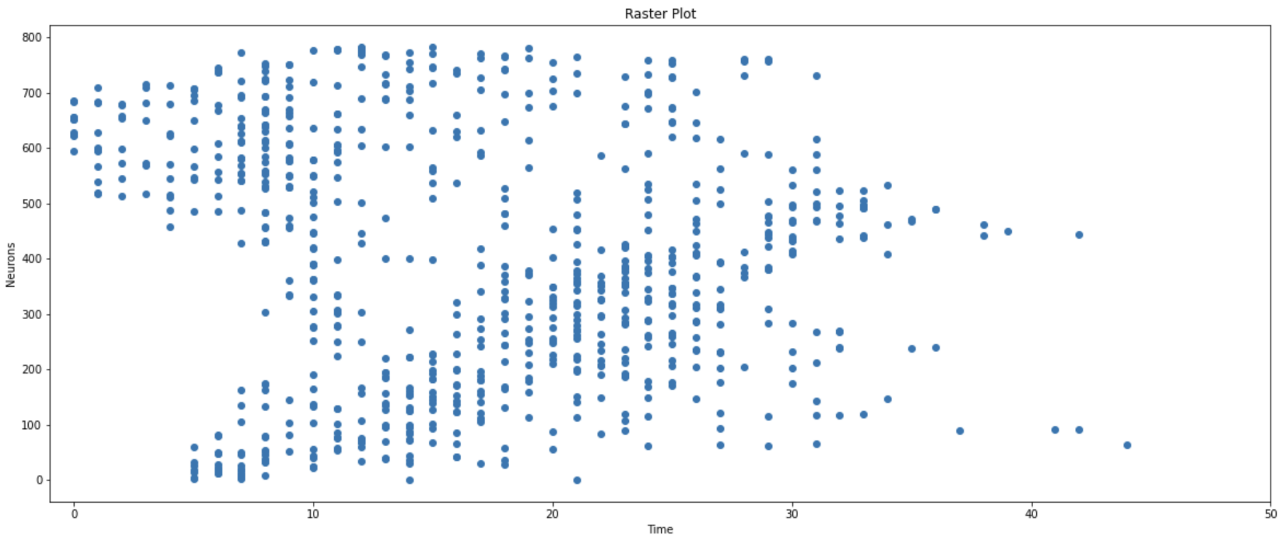

Figure 7: TTFS encoding of an dog image

We trained the five filters with the same parameters as the last experiment. Now let's see the activity of neurons with padding zero:

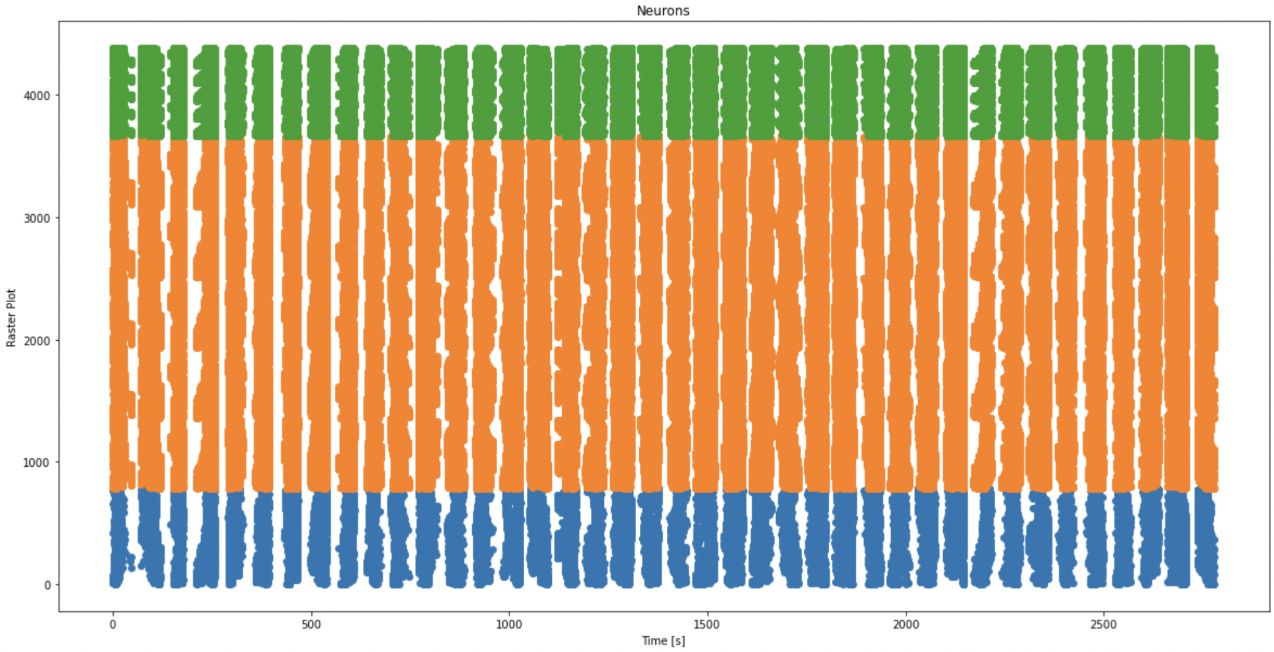

Figure 8: Raster Plot

As you see the activity are different, so we have different features, Let's see the five filters that got learned thorough the simulation:

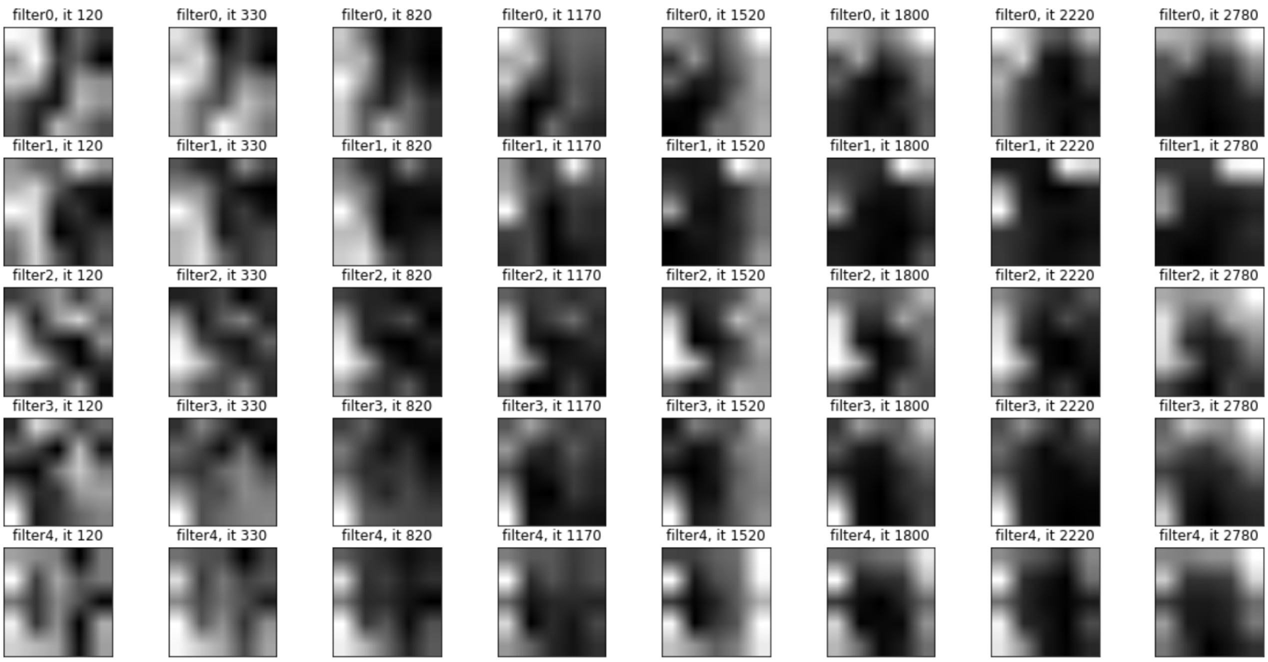

Figure 9: Matrices of weights

So the weights are learned and they are different with face images features, therefore this is the motivation for us to do classification.

5 Classification using RSTDP

Reward-modulated Spike-Timing-Dependent Plasticity (R-STDP) is a learning method for Spiking Neural Network (SNN) that makes use of an external learning signal to modulate the synaptic plasticity produced by Spike-Timing-Dependent Plasticity (STDP). Combining the advantages of reinforcement learning and the biological plausibility of STDP, we can learn features by our selective neurons in the output layer.
codes are in the jupyter network, but results are not finalized yet.

6 Learning with P ganglion cells

Codes and Results: To Do