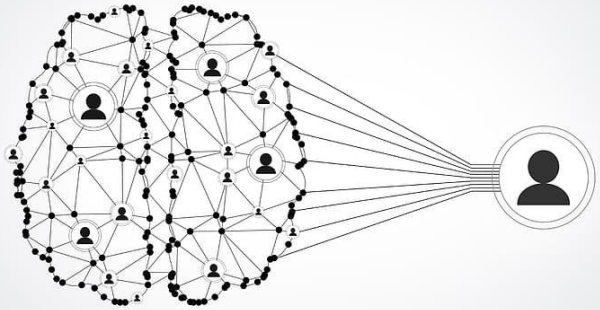
Distilling Spikes: Knowledge Distillation in Spiking Neural Networks

Ravi Kumar Kushawaha, Saurabh Kumar,
Biplab Banerjee, Rajbabu Velmurugan

Indian Institute of Technology Bombay, India



Poster no. - 1254

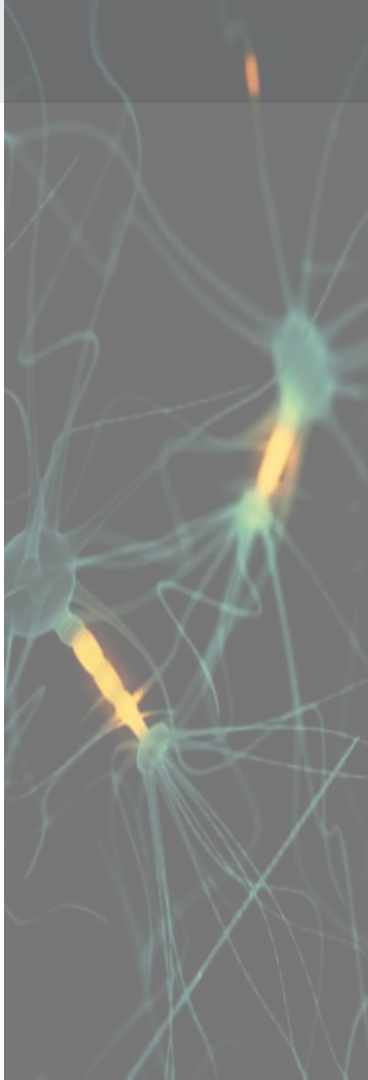
Introduction

01 | A biological neuron in the brain only fires for a short duration when provided with stimulus

02 | The neurons mostly remain inactive and hence require significantly less energy

03 | These pulses are called spikes, and spiking neural networks attempt to model this behavior

04 | SNNs required for developing more efficient computing architectures



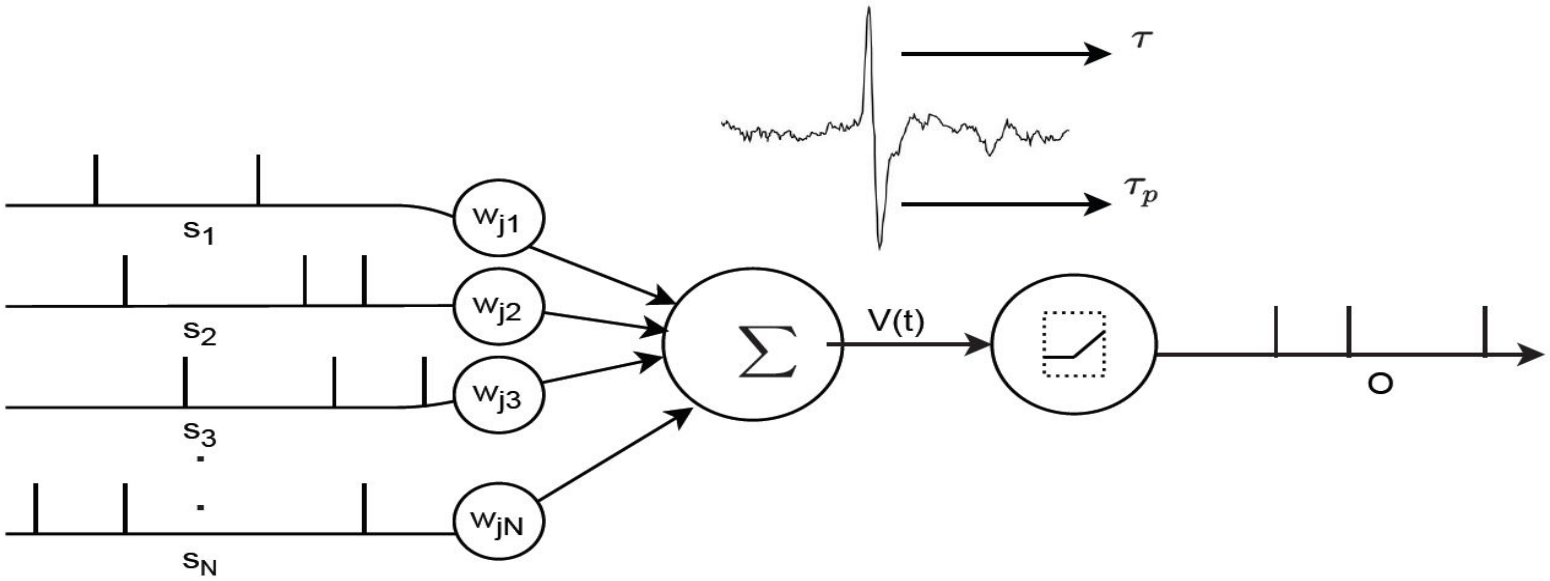


Figure: An illustration of the working of a spiking neuron

Motivation

01 | SNNs are energy-efficient neural models that benefit from deeper architectures like ANNs

02 | Model compression technique to transfer the learning of a large machine learning model to a smaller model with minimal loss in performance

03 | Provide high performance of deeper models while adhering to physical constraints of the available neuromorphic hardware

04 | No prior existing work on knowledge distillation for temporal data

05 | We propose techniques for knowledge distillation in spiking neural networks for the task of image classification

Our contribution

01 | The first-ever method to distill knowledge from a large SNN model trained for image classification

02 | A novel training strategy and multiple objective functions

03 | A multistage knowledge distillation procedure suited for SNNs using an intermediate Teacher Assistant

04 | Demonstrate the effectiveness of the proposed approach by thorough experiments

Proposed Method

Within each layer, SNNs have an extra dimension of time to represent the spike trains, since node values are not scalar

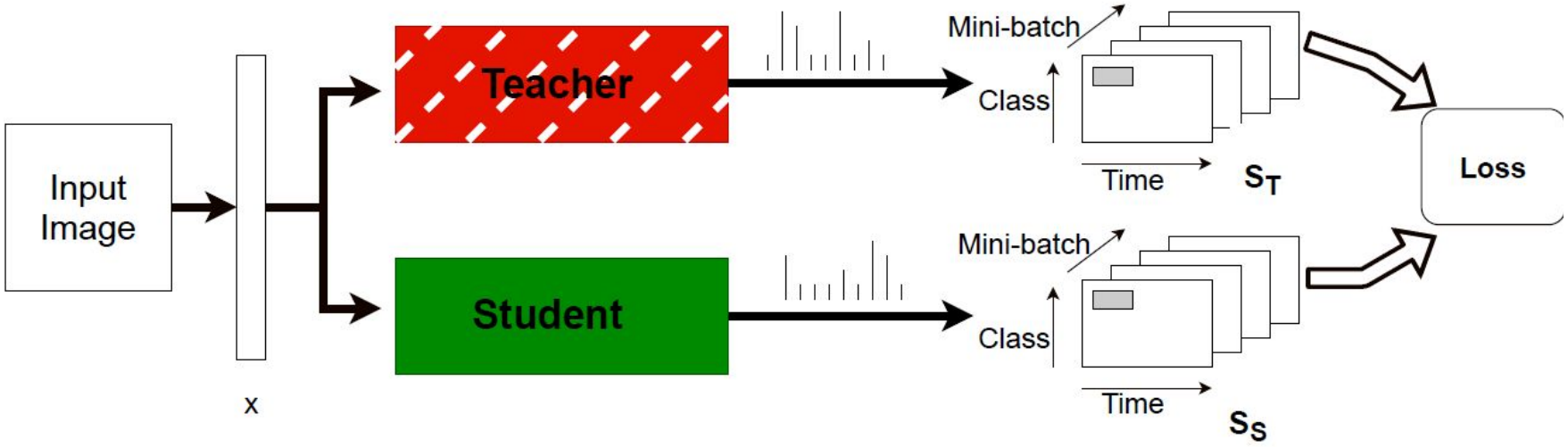
2-D input image flattened to a 1-D vector, and fed into the input layer as constant spike trains

Post-synaptic spikes generated by the input layer are transmitted to the intermediate layers

Input neurons and the output neurons of intermediate layers are densely connected

Output layer uses the spike train from the penultimate layer to generate the final output spike train for classification

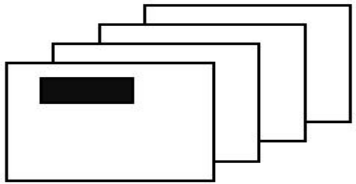
Training methodology



- We first train a teacher SNN which is then used in KD for a student network.
- Given an input image, the weights of teacher SNN are frozen while only the student SNN is trained.
- The KD process involves training this two-stream setup with the proposed loss functions on the post-synaptic spike patterns of the Teacher and Student SNNs

Loss function

- The 3-D tensor (time x classes x mini-batch size) is referred as spiking activation tensor (SAT)
- Losses are calculated by comparing the SATs of both teacher and student model
- L1, L2, KL Divergence loss computed on entire tensors and sliding window losses for L1, L2
- $$L_{sLm} = \sum_{k \epsilon b} \sum_{j \epsilon c} \sum_{i \epsilon t} ||\mathcal{S}_T[i : i + \Delta; j; k] - \mathcal{S}_S[i : i + \Delta; j; k]||_m$$



Results

Classification Accuracy

Table I : Classification accuracy of individual networks trained separately on three datasets

Dataset	MNIST	F-MNIST	CIFAR10
Teacher	98.35	89.72	45.43
TA	98.17	89.4	45.98
Student	98.00	88.64	42.9

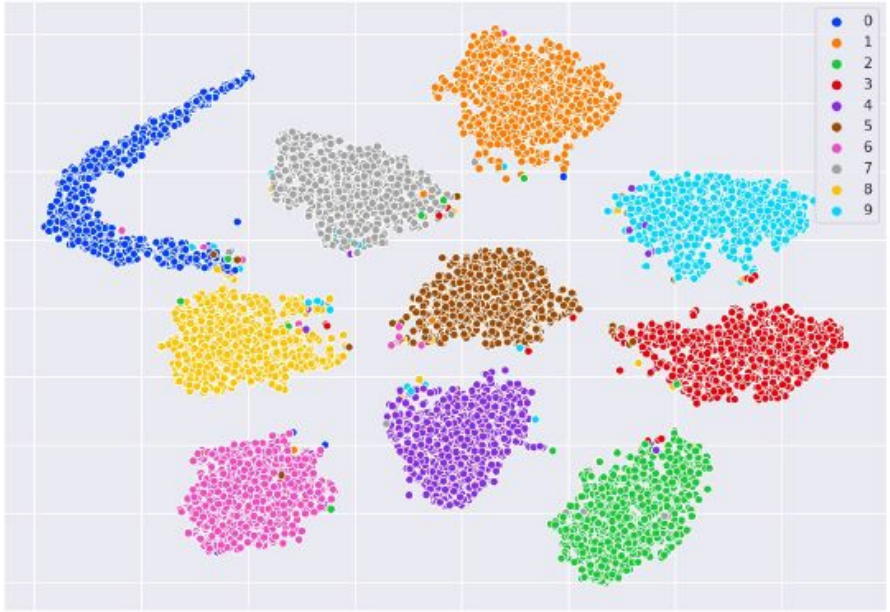
Table III: Classification accuracy using intermediate TA network for KD

Dataset	MNIST	F-MNIST	CIFAR10
Teacher	98.35	89.72	45.43
T $\rightarrow$ TA	98.36	89.82	45.33
T $\rightarrow$ S	97.46	88.30	41.28
T $\rightarrow$ TA $\rightarrow$ S	**97.56**	**88.74**	**42.38**

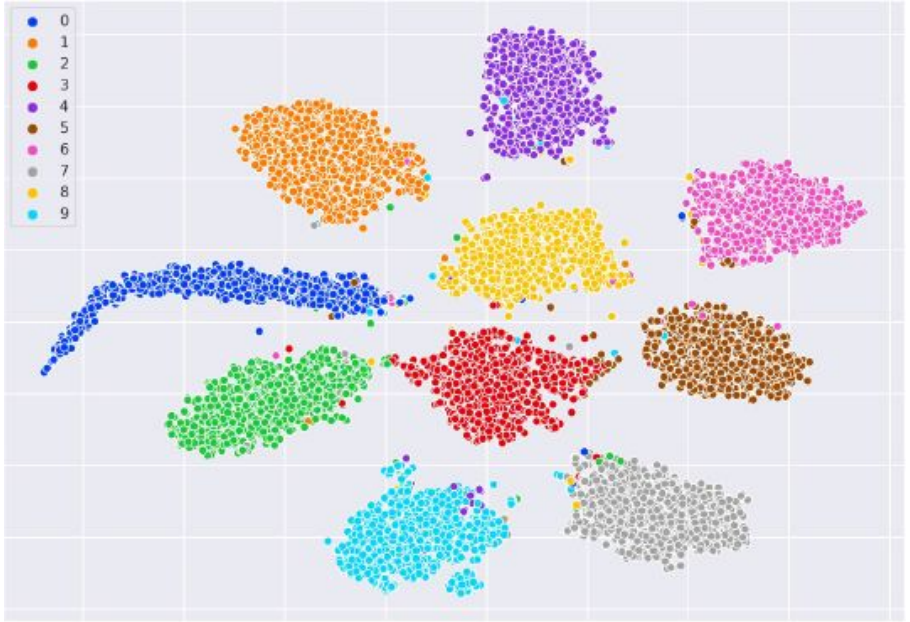
Table II : Performance comparison of Student SNNs with knowledge distilled from the Teacher model using individual components of the proposed loss function

Dataset	MNIST	F-MNIST	CIFAR10
Teacher	98.35	89.72	45.43
Full L1 (L_{L1})	96.20	86.99	37.90
Full L2 (L_{L2})	96.80	87.50	38.70
Full KL (L_{KL})	97.36	88.15	39.21
Sliding L1 (L_{sL1})	96.09	87.28	38.31
Sliding L2 (L_{sL2})	96.29	87.08	38.89
Proposed	**97.46**	**88.30**	**41.28**

t-SNE plots



(a) T → S



(b) T → TA → S

Figure: TSNE plots for MNIST for knowledge distillation with and without the intermediate TA network

Conclusion

01 | SNNs are energy-efficient neural models that benefit from deeper architectures like ANNs

02 | Multistep distillation strategy offers further improvement in performance by using an intermediate TA network

03 | The proposed techniques and objective functions allow an effective spike distillation in SNNs

04 | Practical realization of large SNN models by providing high performance of deeper models

Thank you.

Email id: rkkush2397@gmail.com
LinkedIn : https://www.linkedin.com/in/ravi-kumar-kushawaha-224950121/