First-spike based visual categorization using reward-modulated STDP

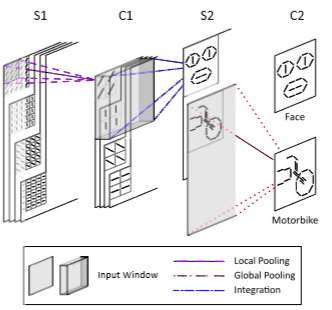
Mozafari, M., Kheradpisheh, S. R., Masquelier, T., Nowzari-Dalini, A., & Ganjtabesh, M. (2018). First-Spike-Based Visual Categorization Using Reward-Modulated STDP. IEEE Transactions on Neural Networks and Learning Systems (https://doi.org/10.1109/TNNLS.2018.2826721).

Introduction:

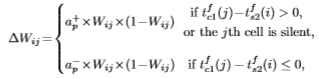
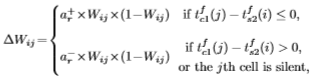
* Problem
  + Current STDP-based SNNs is that they are unsupervised – require external classifier after; classifier often *frame-based*
  + STDP good for frequently occurring features, but bad for “rare but diagnostic” features for classification
* Solution
  + Utilise RL to train highest layer (C2) of network to perform categorisation, using simple features extracted by previous layers
  + RL requires measures to reduce overfitting
* Context
  + Object categorisation from image
  + Datasets: Caltech Face/Motorbike, ETH-80, NORB
  + Latency-to-intensity coding
  + Feedforward SNN as in Masquelier and Thorpe …
* Result & Significance
  + Outperforms standard STDP with basic classifiers, and shallow CNN on greyscale image
  + R-STDP demonstrates plasticity to label changes
* Evaluation
  + Network is inefficient for many categories:
    - Requires k x m (#features x #categories) grids in S2 layer
    - Would need many grids in S2 layer (each grid corresponds to feature for specific category) – if we have say 100 output categories (50 x/y), need 100 x k grids of S2 neurons
    - Would prefer for classifying layer to utilise same feature pool
  + Doesn’t include temporal dynamics as static images
    - Therefore can use IF and first spike dynamics
    - Not applicable to continuous time

Method:

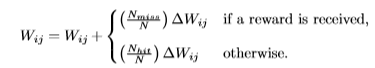
* One spike per neuron in network
* Network used is 4 layer SNN:
  + S1:
    - Grid of Gabor filters selective to orientation; spike latency inversely proportional to saliency (?) of edge
  + C2:
    - Local pooling of S1 to perform position invariance; propagates first spike
  + S2:
    - *n* 2D grids of neurons (*n* *= k x m; k =* #features, m = #categories)
    - Local pooling of C1
    - IF neurons equipped with R-STDP, reward/punishment determined by output of C2 (decision layer)
    - Weight-sharing used, means features learnt across grid
    - WTA used, introduces competition
  + C2:
    - Global pooling of *single S2 neuronal grid* (i.e. a single feature)
    - Therefore *n* neurons in C2 layer
    - Divide layer into *m* groups corresponding to labels
    - First spiking neuron = predicted label
    - If correct decision, reward -> LTP, if incorrect, punishment -> LDP in S2 layer



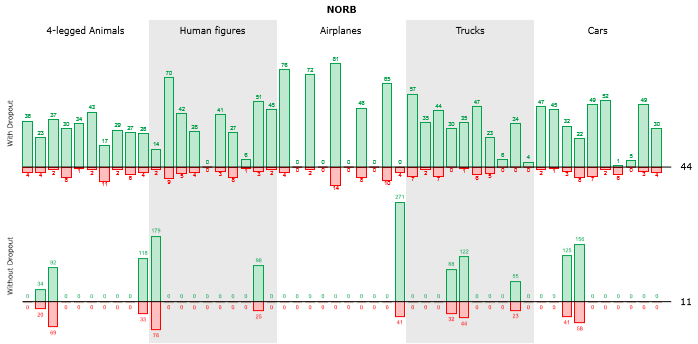
* R-STDP rule:
  + Independent of time
  + Reward/punishment magnitude modulated by ar+, ap+ > 0; ar-, ap- < 0
  + Reward/punishment weight updates shown below:



* Avoiding Overfitting
  + Adaptive Learning Rate (ar, ap)
    - Needed to ensure as learning progresses and punishments become more sparse, their effect remains balanced with reward
    - i.e. as classification gets better, inc strength of punishment, dec strength of reward
    - For each training iteration:

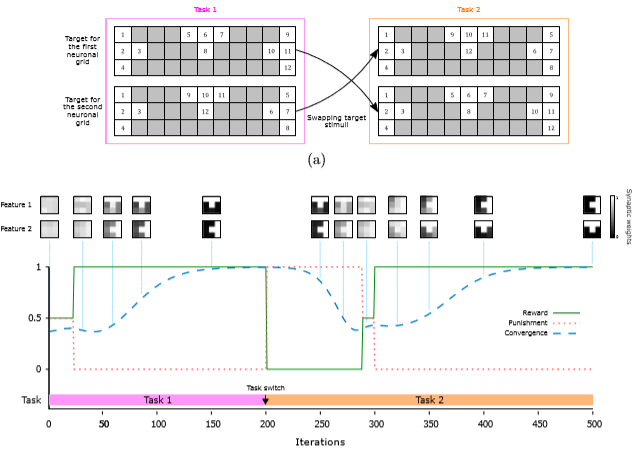


* + Dropout
    - C2 neurons temporarily off w p = pdrop (0.4-0.5 best performance)
    - Inc involvement of neurons (see below)

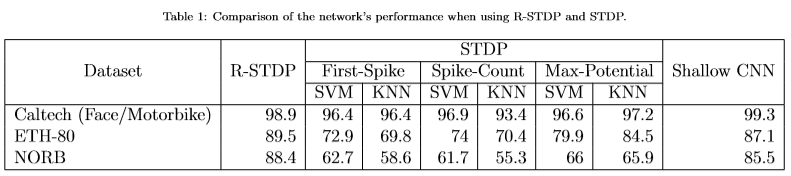


Results:

* Plasticity:



* Classification Accuracy:



* Feature learning:

