End-of-Summer 2015 Report

Justin Skycak

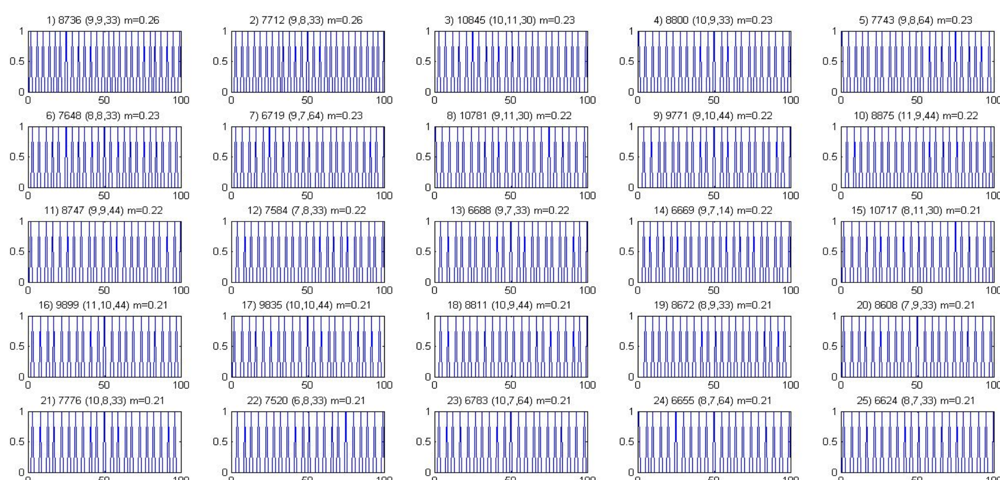
jpskycak@gmail.com

The goal of my summer project was to use PetaVision to observe oscillations in spiking V1 neurons with a leaky integrator error layer.

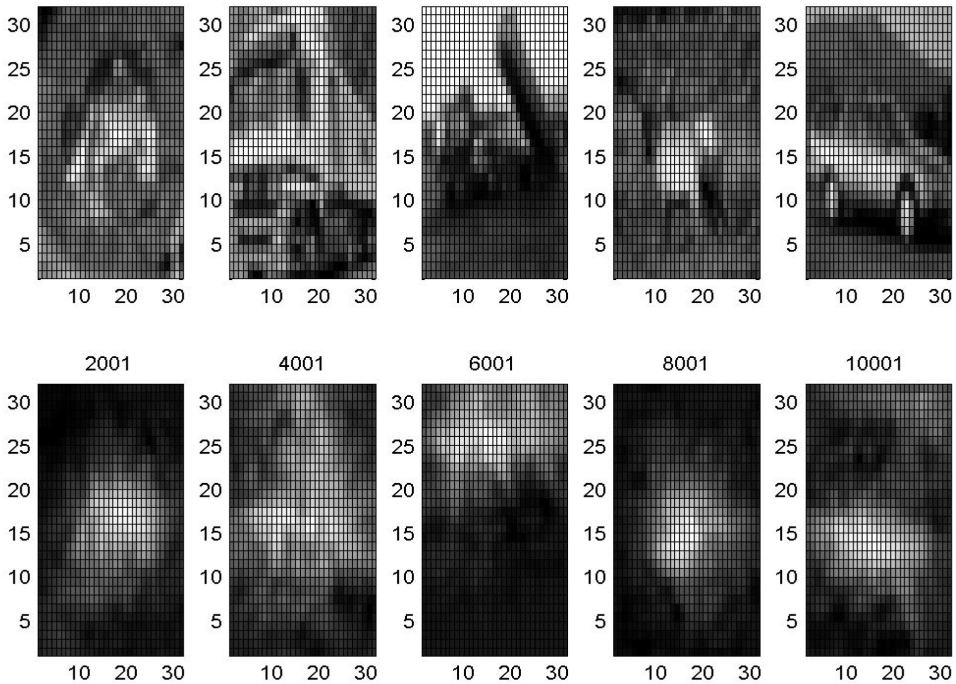
I hypothesized that when a spiking neuron became active, it would over-represent its feature in the reconstruction, thus turning off its error-driven input. As the neuron remained inactive, the neuron's representation would decay until the error-driven input was large enough to activate the neuron again. Many repetitions of this cycle of activity and inactivity would then appear as oscillations in the neuron's spiking rate.

Spiking was enabled through adjusting a neuron's `selfInteract` term. Normally, `selfInteract` was treated as a boolean variable that, if set to true, would feed a neuron's own activity into the neuron's input. Thus, a neuron with `selfInteract = true` would drive itself.

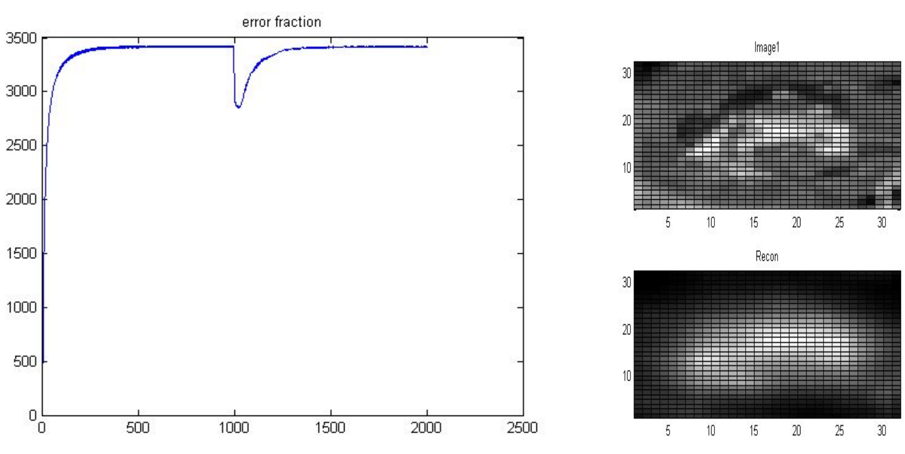
However, by changing `selfInteract` to a float and setting it to a sufficiently negative value, a neuron could be made to inhibit itself. Using a hard threshold and setting a neuron's `selfInteract` term to a negative value (with the magnitude depending on the threshold), a neuron would "reset" itself whenever it became active. Below are activities of 25 different neurons, over 100 timesteps:



However, although the neurons were indeed spiking, their spike rates did not oscillate, and reconstructions were of poor quality:

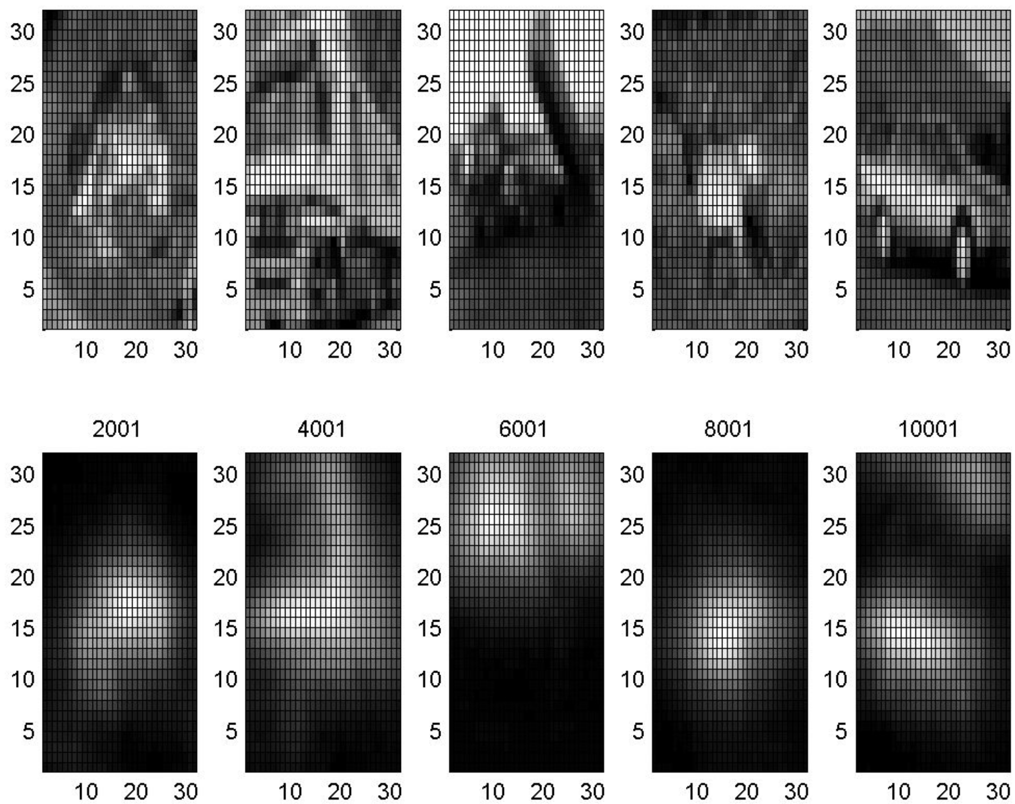


To test whether the lack of oscillations was caused by a representation strength deficiency, I increased the V1 representation strength so that error fraction became very large:

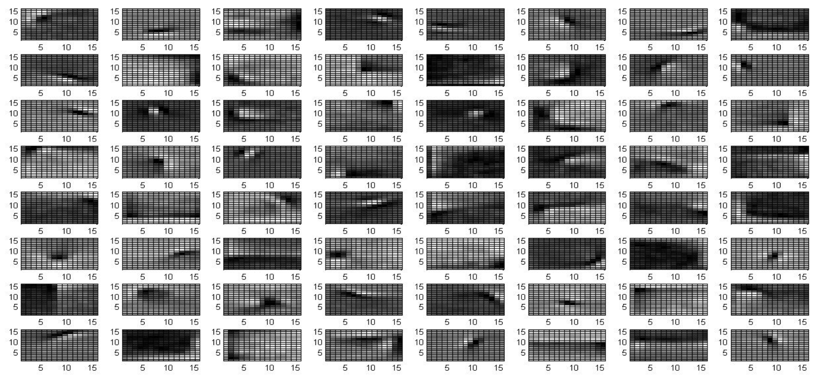


Although the reconstructions became more constant and less wispy, they were still of low quality, and there were no spike-rate oscillations.

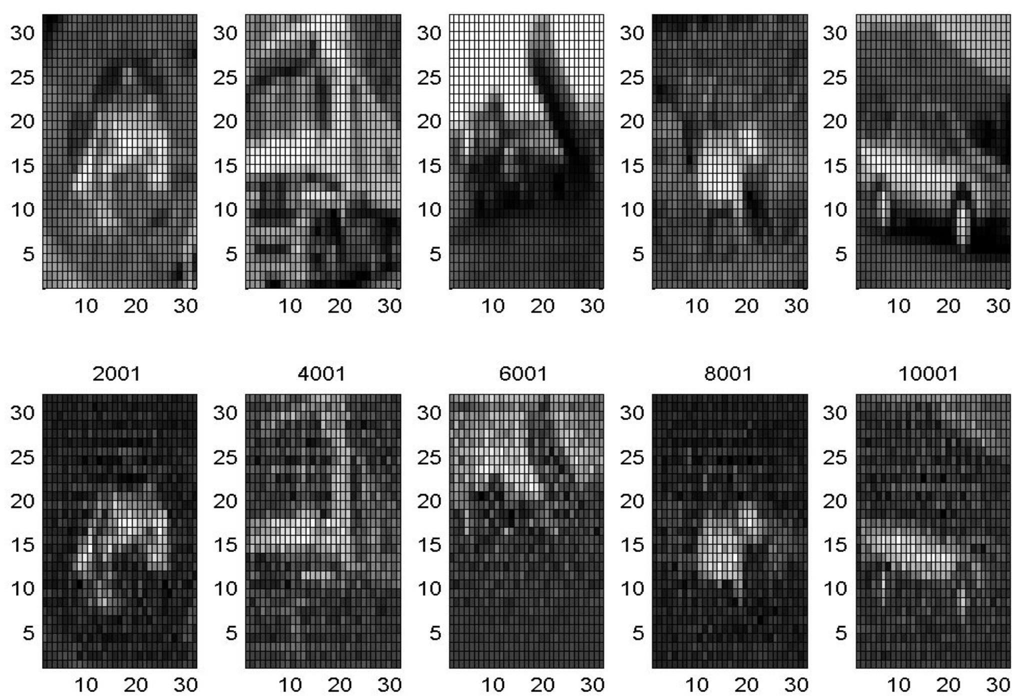
So, I began troubleshooting. I thought that reconstructions formed by using a non-integrating error layer and combining successive instantaneous reconstructions over several timesteps should look similar to reconstructions formed by using an integrating error layer. Indeed, they were (compare the bottom-left reconstruction below to the reconstruction in the previous figure).



Up until this point, all trials involved trained weights:



Interestingly, however, the quality of reconstructions obtained by using a non-integrating error layer and combining instantaneous reconstructions improved when random weights were used:



Although using random weights significantly improved reconstruction quality when using a non-integrating error layer and combining instantaneous reconstructions, random weights did not improve reconstruction quality when using an integrating error layer.