

ПОСТРОЕНИЕ ОТОБРАЖЕНИЯ «АКТИВНОСТЬ НЕЙРОНОВ – КЛАСС» В СВЕРТОЧНОЙ СПАЙКОВОЙ НЕЙРОННОЙ СЕТИ С ОБУЧЕНИЕМ ПО ПРАВИЛУ STDP

А. С. Тощев^[0000-0003-4424-6822]

Казанский (Приволжский) федеральный университет, г. Казань, Россия

atoschev@kpfu.ru

Аннотация

Исследована задача построения отображения между спайковой активностью сверточной спайковой нейронной сети и классом входного изображения. Такая задача появляется после обучения сети без учителя: сверточный слой формирует событийное представление изображения, но сама сеть не содержит готового механизма выдачи метки класса. В работе проведена проверка, насколько информативны счетчики спайков сверточного слоя после обучения по правилу STDP (пластичность, зависящая от времени возникновения импульсов, – биологически правдоподобное правило обучения для импульсных (спайковых) нейронных сетей), и оценено качество линейного считывающего классификатора при увеличении числа обучающих предъявлений.

Эксперименты проведены на наборе MNIST – стандартном наборе изображений рукописных цифр. Использована однослойная архитектура: один сверточный слой с 32 картами признаков, ядром 5×5 и выходной размерностью $32 \times 24 \times 24$, что соответствует 18432 нейронам LIF (модель «интегратор с утечкой и порогом»). Входные изображения кодировались детерминированным пуассоновским кодировщиком; мультипликатор пуассоновского кодирования был равен 0.006. Длительность предъявления (время демонстрации объекта сети) составляла 100 тактов. Обучение выполнялось по правилу STDP с нуля, без возобновления из ранее сохраненного состояния. Для оценки использовался протокол сбора счетчиков спайков на 10000 обучающих и 10000 тестовых изображениях.

В основном эксперименте при 15000 обучающих предъявлениях STDP достигнута точность 0.8883. Базовый способ по карте меток, близкий к методу Diehl и Cook, в той же конфигурации дал около 0,48. Покрытие калиброванной карты

меток составило 219 из 18432 нейронов, средняя активность – 4146.9416 спайка на изображение. Соседняя точка масштаба при 10000 обучающих предъявлениях дала точность 0.8876. Полученные результаты показывают, что выбранный способ считывания активности существенно влияет на итоговое качество классификации: линейный считывающий классификатор по полному вектору счетчиков сети извлекает распределенную информацию, которая теряется при простом назначении меток отдельным нейронам.

Ключевые слова: *спайковые нейронные сети, сверточные спайковые сети, считывающий классификатор, счетчики спайков, классификация изображений.*

ВВЕДЕНИЕ

Спайковые нейронные сети (СНС) представляют собой класс моделей, в которых вычисления и передача информации задаются во времени, а базовой единицей взаимодействия между нейронами выступает импульс – кратковременное дискретное событие. В отличие от традиционных нейронных сетей, работающих с непрерывными активациями, СНС опираются на временную структуру сигналов и редкие события, что рассматривается как одно из принципиальных отличий третьего поколения нейросетевых моделей [1, 2].

Такой принцип организации делает СНС естественно близкими к биологическим нейронным системам и одновременно открывает путь к событийно-управляемым вычислениям, в которых вычислительная стоимость определяется не только числом арифметических операций, но и количеством сгенерированных импульсов. Именно поэтому СНС и нейроморфные системы рассматриваются как перспективное направление для энергоэффективных вычислений, особенно при использовании специализированного аппаратного обеспечения [3]. Вопрос энергопотребления нейронных вычислений и нейроморфных архитектур также рассматривается в работах, посвященных влиянию нейросетевых моделей на вычислительные ресурсы и энергоэффективность [4, 5].

Для задач компьютерного зрения интерес представляют сверточные спайковые нейронные сети. В них локальные признаки изображения извлекаются сверточными связями, а вычисление внутри слоя остается событийным. Такие

модели важны не только как инженерный вариант классификатора, но и как способ изучать, насколько далеко можно продвинуться при обучении без обратного распространения ошибки.

В статье [6] рассмотрено преобразование статического изображения во временной вход для СНС. Настоящая работа продолжает эту линию, но переносит акцент на следующий этап: построение обратного отображения от импульсной активности к классу. Иначе говоря, если задача [6] состояла в том, как подать изображение в СНС, то здесь рассматривается вопрос, как считать ответ из обученной спайковой сети.

При этом после обучения без учителя возникает отдельная, иногда недооцененная задача: как связать активность нейронов с меткой класса? Сеть может научиться реагировать на штрихи, края и локальные фрагменты цифры. Но класс «0», «1» или «7» не записан в ее весах напрямую. Поэтому нужен считывающий классификатор – компонент, который получает признаки по активности нейронов и возвращает метку класса.

Таким образом, цель настоящей работы – описать и экспериментально проверить отображение «активность нейронов – класс» для однослойной сверточной СНС, обученной на MNIST (датасет изображений рукописных цифр) по правилу STDP (пластичность, зависящая от времени возникновения импульсов, – биологически правдоподобное правило обучения для импульсных (спайковых) нейронных сетей). Основные задачи состоят в следующем: описать воспроизводимый протокол построения признаков по счетчикам спайков сверточного слоя нейронов LIF; зафиксировать схему линейного считывающего классификатора по стандартизованным счетчикам; выполнить сравнение с базовым методом назначения меток нейронам, близким к подходу Diehl и Cook [7].

СВЯЗАННЫЕ РАБОТЫ И ОСНОВНЫЕ ПОНЯТИЯ

Diehl и Cook показали [7], что СНС с STDP, латеральным торможением и адаптивными порогами может решать задачу распознавания цифр без предъявления меток во время обучения. В их подходе после обучения каждому нейрону назначается метка того класса, на объектах которого этот нейрон в сред-

нем активируется сильнее. Затем класс нового изображения определяется по активности нейронов с назначенными метками. Для настоящей работы этот метод принципиален как естественная базовая линия: он показывает, что прямое назначение меток нейронам не всегда достаточно для извлечения всей информации из распределенной активности.

Сверточные варианты СНС развивают ту же идею для изображений: локальные фильтры позволяют нейронам реагировать не на весь вход сразу, а на фрагменты изображения. В работе [8] показано, что STDP в сверточных СНС может приводить к формированию признаков различной сложности и использоваться для распознавания объектов. В нашей работе использована более узкая постановка: один сверточный слой, MNIST и анализ считывающего классификатора по счетчикам спайков.

MNIST – это стандартный набор изображений рукописных цифр. Он содержит 70000 изображений размером 28×28 пикселей, относящихся к 10 классам [9]. В экспериментах применялось стандартное деление: обучающая часть использовалась для STDP-обучения и сбора признаков считывающего классификатора, а тестовая часть – для итоговой оценки качества.

STDP (времязависимая синаптическая пластичность) – это правило изменения синаптических весов, зависящее от относительного времени импульсов пресинаптического и постсинаптического нейрона. Биологическая основа такого механизма описана в работах [10, 11]. В вычислительных моделях STDP часто используется для обучения спайковых сетей без учителя.

Нейрон LIF (leaky integrate-and-fire neuron, интегратор с утечкой и порогом) – это модель нейрона с утечкой мембранного потенциала и генерацией импульса при достижении порога. Такая модель проще биофизически детальных моделей, но удобна для больших вычислительных экспериментов [12].

МАТЕРИАЛЫ И МЕТОДЫ

Экспериментальная выборка. Экспериментальная последовательность этапов была построена на библиотеке BindsNET, предназначенной для моделирования спайковых нейронных сетей и интегрированной с PyTorch [13]. Использовался набор MNIST. Изображения имеют размер 28×28 пикселей и представлены в оттенках серого. Число классов равно 10.

В серии экспериментов были использованы два разных числа обучающих предъявлений для STDP: 10000 обучающих предъявлений как соседняя точка масштаба и 15000 обучающих предъявлений в основном эксперименте. После обучения выполнялся протокол оценки по счетчикам: счетчики спайков собирались на 10000 изображений обучающей части и 10000 изображений тестовой части. Тестовые изображения не использовались для обучения весов STDP и расчета параметров стандартизации.

Архитектура сверточной СНС. Сеть имела один входной и один сверточный слой нейронов LIF. Входное изображение подавалось как последовательность спайковых кадров. Сверточный слой имел 32 выходные карты признаков и ядро 5×5 .

Размерность выхода сверточного слоя рассчитывалась по формуле

$$H_{\text{out}} = W_{\text{out}} = H - k + 1,$$

где $H = 28$ – высота входного изображения, $k = 5$ – размер ядра, шаг равен 1, дополнение нулями не применялось. Поэтому $H_{\text{out}} = W_{\text{out}} = 28 - 5 + 1 = 24$.

Общее число нейронов LIF в сверточном слое определяется формулой

$$N_{\text{LIF}} = C_{\text{out}} H_{\text{out}} W_{\text{out}} = 32 \cdot 24 \cdot 24 = 18432,$$

где $C_{\text{out}} = 32$ – число карт признаков, $H_{\text{out}} = W_{\text{out}} = 24$ – пространственный размер выхода сверточного слоя.

Сеть обучалась по схеме: сверточный слой и STDP-обучение. Обратное распространение ошибки при обучении сверточных весов не применялось. Использовались адаптивный порог, возбуждающе-тормозной режим и нормировка весов (см. табл. 1).

Табл. 1. Конфигурация основного эксперимента.

Параметр	Значение
Архитектура	сверточная СНС
Слой	сверточный LIF-слой
Число карт признаков	32
Размер ядра	5×5
Размер выхода сверточного слоя	$32 \times 24 \times 24$
Число LIF-нейронов	18432

Набор данных	MNIST
Вычислительное устройство	графический ускоритель CUDA
Кодировщик	детерминированный пуассоновский
Мультипликатор пуассоновского кодирования	0.006
Длительность предъявления	100 тактов
Число обучающих предъявлений STDP	15000
Адаптивный порог	включен
Постоянная адаптации порога	10000
Прибавка к порогу после импульса	0.05
Возбуждающе-тормозной режим	включен
Коэффициент возбуждения	22.5
Коэффициент торможения	120
Нормировка весов	включена
Целевое значение нормировки весов	12.5
Режим обучения	обучение с нуля, без возобновления из сохраненного состояния

Кодирование входа. Для преобразования изображения в поток спайков применялся пуассоновский кодировщик [6]. Пуассоновское кодирование переводит яркость пикселя в вероятность генерации импульса за такт моделирования: чем выше значение пикселя, тем выше ожидаемая частота импульсов. В этой серии экспериментов использовался детерминированный вариант кодировщика. Это важно для воспроизводимости: при одинаковых изображении и параметрах входной поток от эксперимента к эксперименту не изменяется случайно.

Интенсивность кодирования задавалась мультипликатором пуассоновского кодирования 0.006. Длительность предъявления одного изображения составляла $T = 100$ тактов.

Сбор счетчиков спайков. После STDP-обучения веса сети фиксировались. Затем каждое изображение прогонялось через сеть, а для каждого нейрона LIF сверточного слоя вычислялось число спайков за окно предъявления.

Для изображения x и нейрона j счетчик спайков определялся по формуле

$$c_j(x) = \sum_{t=1}^T s_j(t, x),$$

где $s_j(t, x) \in \{0, 1\}$ – факт спайка нейрона j в момент времени t , $T = 100$ – длительность предъявления изображения.

Вектор счетчиков задавался формулой

$$c(x) = (c_1(x), c_2(x), \dots, c_{18432}(x)).$$

Вектор $c(x)$ использовался как исходное описание активности сверточного слоя.

В программной реализации сбор счетчиков поддерживает сверточный монитор: выход вида $[T, B, C, H, W]$ приводится к плоскому виду $[T, B, N]$, где $N = 18432$. Это позволяет использовать один и тот же принцип считывающего классификатора для полносвязного и сверточного скрытого слоя.

Базовый считывающий классификатор по карте меток. В качестве базовой линии был использован подход, близкий к методу Diehl и Cook [7]. После STDP-обучения для каждого нейрона анализировалась его средняя активность на обучающих изображениях разных классов. Нейрону назначалась метка того класса, для которого его средняя активность была максимальной. При классификации нового изображения учитывалась активность нейронов с назначенными метками.

Такой способ хорошо согласуется с идеей обучения без учителя: веса формируются без меток, а метки используются только на этапе калибровки. Однако для рассматриваемой сверточной сети этот вариант оказался слабее линейного считывающего классификатора по полному вектору счетчиков. В нашей конфигурации точность базового метода составила около 0.48. Поэтому далее он рассматривается как контрольная точка, показывающая ограниченность прямого назначения меток нейронам.

Линейный считывающий классификатор по стандартизованным счетчикам. Лучшим вариантом оценки в проведенной серии экспериментов стал линейный считывающий классификатор по стандартизованным счетчикам. Его

идея состоит в том, что каждый нейрон описывается числом спайков, затем признаки логарифмируются, стандартизируются и подаются на линейную регрессию.

Логарифмирование счетчиков выполнялось по формуле

$$u_j(x) = \log(1 + c_j(x)).$$

Затем параметры нормализации рассчитывались только на обучающей части оценки по счетчикам

$$z_j(x) = \frac{u_j(x) - \mu_j}{\sigma_j + \varepsilon},$$

где μ_j и σ_j – среднее и стандартное отклонения признака j по обучающей части, ε – малое число для защиты от деления на нуль. Идея такой стандартизации соответствует обычной z-score нормализации признаков [14].

После преобразования обучалась линейная регрессия

$$\hat{y}(x) = \arg \max_k (Wz(x) + b)_k,$$

где W и b – параметры линейного классификатора, $k \in \{0, \dots, 9\}$ – номер класса. В проведенной реализации линейная регрессия обучалась с функцией потерь перекрестной энтропии оптимизатором Adam [15], коэффициентом регуляризации 0,0001, числом эпох 25 и размером пакета 512.

Оцениваемые характеристики. Основная метрика качества – точность классификации

$$\text{Accuracy} = \frac{N_{\text{correct}}}{N_{\text{test}}},$$

где N_{correct} – число верно классифицированных тестовых изображений, N_{test} – общее число тестовых изображений.

Кроме точности фиксировались две диагностические величины. Первая – это среднее число спайков на изображение

$$S_{\text{avg}} = \frac{1}{M} \sum_{i=1}^M \sum_{j=1}^{N_{\text{LIF}}} c_j(x_i),$$

где M – число изображений в измеряемой части, N_{LIF} – число нейронов LIF в сверточном слое. Эта величина трактуется как прокси активности, а не как физическое энергопотребление.

Вторая величина – это покрытие карты меток, т. е. число нейронов, которым при калибровке была назначена метка класса

$$\text{Coverage} = \frac{N_{\text{assigned}}}{N_{\text{LIF}}},$$

где N_{assigned} – число нейронов, которым при калибровке была назначена метка класса. Эта величина не равна точности классификации. Она показывает, какая часть нейронов получила явное соответствие классу при построении карты меток. Для основного считывающего классификатора используется полный вектор счетчиков спайков, а не только нейроны с назначенной меткой.

РЕЗУЛЬТАТЫ

Основной эксперимент был выполнен при 15000 обучающих предъявлениях в режиме STDP. Расчеты выполнялись на графическом ускорителе CUDA. Полная длительность эксперимента составила 2:03:09, или 7389 с.

После завершения обучения были сохранены итоговая модель, контрольное состояние после обучения, карта меток и журналы активности. Для оценки использовалась единая последовательность этапов: сначала обучение STDP, затем калибровка карты меток, сбор счетчиков спайков и обучение линейного считывающего классификатора.

Табл. 2. Итоговые метрики основного эксперимента.

Показатель	Значение
Число обучающих предъявлений STDP	15000
Точность	0.8883
Средняя активность, количество спайков на изображение	4146.9416
Прокси активности, количество спайков на изображение	4146.9416
Нейроны с назначенной меткой класса	219
Всего LIF-нейронов	18432
Доля назначенных нейронов	1.188 %
Вычислительное устройство	графический ускоритель CUDA

Из таблицы видно, что полный вектор счетчиков спайков сверточного слоя содержит достаточно информации для линейного разделения классов MNIST на уровне точности 0.8883. Что немаловажно, явную метку при калибровке получили только 219 нейронов из 18432, т. е. около 1.188%. Значит, качество считывающего классификатора не стоит сводить к простому голосованию небольшой группы «подписанных» нейронов. Линейная регрессия использует гораздо более широкое распределенное описание активности.

Сравнение способов построения считывающего классификатора. В табл. 3 проведено сравнение двух способов сопоставления активности сверточной СНС с классами MNIST. Первый способ – это базовый считывающий классификатор по карте меток, близкий к подходу Diehl и Cook [7]. Второй способ – линейный считывающий классификатор по стандартизованным счетчикам спайков.

Табл. 3. Сравнение способов построения считывающего классификатора.

Способ построения считывающего классификатора	Используемое представление активности	Точность
Карта меток по средней активности нейронов, близко к Diehl и Cook [7]	Активность нейронов с назначенной меткой класса	около 0.48
Линейный рид-аут по стандартизованным счетчикам	Полный вектор счетчиков спайков сверточного слоя	0.8883

Сравнение показало, что простое назначение метки отдельным нейронам плохо использует распределенную активность сверточного слоя. В основном эксперименте явную метку получили только 219 нейронов из 18432. Если опираться преимущественно на такую карту, то значительная часть полезного сигнала потеряется. Линейный считывающий классификатор работает иначе: он использует полный вектор счетчиков спайков и может учитывать слабые, но совместно информативные реакции большого числа нейронов.

Сравнение соседних точек масштаба. Для оценки влияния числа STDP-предъявлений были сопоставлены две соседние точки масштаба: 10000 и 15000

обучающих предъявлений. Оба эксперимента использовали одну и ту же архитектуру, один кодировщик и один оценочный протокол по счетчикам спайков (см. табл. 4).

Табл. 4. Масштабирование STDP-обучения без возобновления из сохраненного состояния.

Число обучающих предъявлений STDP	Точность	Нейроны с назначенной меткой класса	Всего нейронов	Количество спайков на изображение
10000	0.8876	214	18432	4122.8895
15000	0.8883	219	18432	4146.9416

Переход от 10000 к 15000 предъявлений дал прирост точности на 0.0007. В абсолютных единицах это 0.07 процентного пункта. Покрытие карты меток выросло на 5 нейронов – с 214 до 219. Средняя активность увеличилась с 4122.8895 до 4146.9416 спайка на изображение, т. е. на 24.0521 спайка.

Прирост качества оказался очень малым. По времени же эксперимент при 15000 обучающих предъявлениях занял 2:03:09. На практике это означает, что простое увеличение числа STDP-предъявлений в выбранной однослойной архитектуре уже приближается к зоне малой отдачи. Это не строгий статистический вывод, потому что нет серии по нескольким случайным инициализациям. Но как инженерный сигнал – вполне заметный: масштабирование обучения почти не сдвигает точность.

Что показывает покрытие карты меток. Покрытие карты меток в основном эксперименте равно

$$\text{Coverage} = \frac{219}{18432} \approx 0.01188.$$

В процентах это около 1.188%. На первый взгляд может показаться, что столь малое покрытие должно ограничивать качество. Однако точность 0.8883 указывает на другое: полезная информация распределена в полном векторе счетчиков. Многие нейроны могут не получить устойчивой индивидуальной метки, но их активность все равно влияет на линейное решение через веса считывающего

классификатора. Отсюда следует важное методическое наблюдение: для сверточной СНС после STDP не стоит оценивать качество представления только по числу нейронов с назначенной меткой, нужно проверять считываемость активности отдельным классификатором. В нашей работе таким классификатором выступает линейная регрессия после логарифмирования и стандартизации счетчиков.

ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

Результат 0.8883 на MNIST заметно выше результата базового считывающего классификатора по карте меток, близкого к методу Diehl и Cook [7], который в нашей конфигурации дал около 0.48. Это сравнение важно: оно показывает, что основное улучшение связано не только с обучением сверточного STDP-слоя, но и с тем, как затем считывается его активность.

Результат 0.8883 также заметно выше результата полносвязной СНС из предыдущего этапа серии, где лучший считывающий классификатор на основе TF-IDF и многослойного перцептрона дал величину 0.5467. Прямое сравнение этих чисел нужно делать осторожно, потому что различаются архитектура, число нейронов, длительность предъявления и вычислительная стоимость. Но общий вывод понятен: сверточный слой дает более подходящую структуру признаков для изображений, чем небольшой полносвязный скрытый слой.

Интересно и другое. В варианте [6] лучшим оказался нелинейный считывающий классификатор. В текущей серии экспериментов лучшим стал линейный считывающий классификатор по стандартизованным счетчикам. Это говорит о том, что при переходе к большому сверточному слою простое линейное считывание уже оказывается достаточным для сильного результата на базе MNIST. Вероятно, сама сверточная архитектура формирует более линейно разделимое пространство счетчиков. Это пока гипотеза, ее нужно проверять на других выборках и при разных размерах сверточного слоя.

Почти нулевой прирост при переходе от 10000 к 15000 обучающих предъявлений важен для планирования дальнейших экспериментов. Эксперимент стал дороже по времени, но точность выросла только на 0.0007. С практической точки зрения это похоже на плато: выбранный набор параметров – 32 карты при-

знаков, ядро 5×5 , длительность предъявления 100 тактов, мультипликатор пуассоновского кодирования 0.006, коэффициент торможения 120 и прибавка к порогу после импульса 0.05 – уже извлекает основную пользу из доступного STDP-обучения.

ОГРАНИЧЕНИЯ И БУДУЩАЯ РАБОТА

Проведенные эксперименты имеют несколько ограничений.

Во-первых, основная оценка выполнена на MNIST. Этот набор удобен для отладки и сравнения, но он проще современных задач компьютерного зрения. Поэтому перенос результата на более сложные изображения нельзя считать автоматическим.

Во-вторых, использована однослойная сверточная СНС. Архитектуры с несколькими сверточными слоями могут вести себя иначе: часть признаков станет более абстрактной, а считывающий классификатор может получить другое пространство счетчиков.

В-третьих, сравнение 10000 и 15000 обучающих предъявлений выполнено по соседним экспериментам серии, но без статистики по нескольким случайным инициализациям. Значение прироста 0.0007 нужно трактовать как экспериментальное наблюдение для текущей конфигурации, а не как универсальную оценку насыщения STDP.

В-четвертых, прокси активности совпадает со средним числом спайков на изображение. Это показатель активности модели в программном эксперименте. Он не является измерением энергопотребления аппаратной реализации.

В-пятых, эксперимент при 30000 обучающих предъявлений был прерван раньше срока. Поэтому вопрос о дальнейшем масштабировании остается открытым, хотя имеющаяся пара точек уже показывает слабую отдачу от увеличения числа предъявлений в диапазоне 10000–15000.

Будущая работа должна объединить три направления: проверку архитектурных изменений, анализ устойчивости по нескольким случайным инициализациям и расширение набора способов считывания активности. Это позволит перейти от удачного инженерного результата на MNIST к более общему протоколу построения считывающего классификатора для сверточных спайковых сетей.

ЗАКЛЮЧЕНИЕ

Построено отображение «активность нейронов – класс» для однослойной сверточной спайковой нейронной сети, обученной по правилу STDP. Архитектура включала один сверточный слой нейронов LIF с 32 картами признаков и ядром 5×5 ; выходная размерность составила 18432 нейрона.

Основной эксперимент при 15000 обучающих предъявлениях достиг точности 0.8883 на тестовой части MNIST. Лучшим вариантом стал линейный считывающий классификатор по стандартизованным счетчикам: логарифмированные счетчики спайков, z-score нормализация по обучающей части и линейная регрессия.

Сравнение с базовым способом по карте меток, близким к методу Diehl и Cook [7], показало принципиальную разницу: прямое назначение меток нейронам в этой же конфигурации дает значение около 0.48, тогда как линейный считывающий классификатор по полному вектору счетчиков достигает 0.8883. Это подчеркивает роль выбранного способа считывания активности.

Сравнение с экспериментом при 10000 обучающих предъявлениях показало прирост точности только на 0.0007: с 0.8876 до 0.8883. Покрытие карты меток выросло с 214 до 219 назначенных нейронов из 18432, а средняя активность изменилась с 4122.8895 до 4146.9416 спайка на изображение. Следовательно, для выбранной однослойной сверточной СНС простое увеличение числа STDP-предъявлений уже дает малую отдачу.

Главный практический вывод состоит в том, что считывающий классификатор по полному вектору счетчиков спайков способен извлекать распределенную информацию даже тогда, когда явную метку получает лишь небольшая доля нейронов. Следующий этап серии стоит направить не только на рост числа предъявлений, но и на проверку архитектурных изменений и устойчивость результатов по нескольким случайным инициализациям.

Благодарности

Работа выполнена за счет гранта, предоставленного Академией наук Республики Татарстан образовательным организациям высшего образования, научным и иным организациям на поддержку планов развития кадрового потенциала в части стимулирования их научных и научно-педагогических работников к

защите докторских диссертаций и выполнению научно-исследовательских работ (Соглашение от 22.12.2025 № 12/2025-ПД-КФУ).

СПИСОК ЛИТЕРАТУРЫ

1. *Gerstner W., Kistler W.* Spiking Neuron Models: Single Neurons, Populations, Plasticity. Cambridge: Cambridge University Press, 2002.
2. *Maass W.* Networks of Spiking Neurons: The Third Generation of Neural Network Models // *Neural Networks*. 1997. Vol. 10, No. 9. P. 1659–1671.
3. *Davies M. et al.* Loihi: A Neuromorphic Manycore Processor with On-Chip Learning // *IEEE Micro*. 2018. Vol. 38, No. 1. P. 82–99.
4. *Chertopolokhov V., Devyatkin D., Serov A., Babaev A., Zeleniy A., Kamyshinsky R., Khmelenin D., Gerasimov E., Tsybin I., Trigub A. L., Kashkarov P., Toshchev A., Popov A.* Advancing Neural Networks: Innovations and Impacts on Energy Consumption // *Advanced Electronic Materials*. 2024. Vol. 10, No. 8. Art. 2400029.
5. *Toshev A. et al.* Critical Analysis of Energy Consumption in Neuro-Computational Systems // *IEEE Access*. 2026. Art. 60535.
6. *Тоцев А.С.* Генерация временных сигналов из статических изображений для подачи на спайковые нейронные сети // *Электронные библиотеки*. 2026. Т. 29. № 3. С. 1061–1077.
7. *Diehl P.U., Cook M.* Unsupervised Learning of Digit Recognition Using Spike-Timing-Dependent Plasticity // *Frontiers in Computational Neuroscience*. 2015. Vol. 9. Art. 99.
8. *Kheradpisheh S.R., Ganjtabesh M., Thorpe S.J., Masquelier T.* STDP-Based Spiking Deep Convolutional Neural Networks for Object Recognition // *Neural Networks*. 2018. Vol. 99. P. 56–67.
9. *LeCun Y., Bottou L., Bengio Y., Haffner P.* Gradient-Based Learning Applied to Document Recognition // *Proceedings of the IEEE*. 1998. Vol. 86, No. 11. P. 2278–2324.
10. *Bi G.Q., Poo M.M.* Synaptic Modifications in Cultured Hippocampal Neurons: Dependence on Spike Timing, Synaptic Strength, and Postsynaptic Cell Type // *The Journal of Neuroscience*. 1998. Vol. 18, No. 24. P. 10464–10472.
11. *Caporale N., Dan Y.* Spike Timing-Dependent Plasticity: A Hebbian Learning Rule // *Annual Review of Neuroscience*. 2008. Vol. 31. P. 25–46.

12. *Izhikevich E.M.* Which Model to Use for Cortical Spiking Neurons? // IEEE Transactions on Neural Networks. 2004. Vol. 15, No. 5. P. 1063–1070.
 13. *Hazan H. et al.* BindsNET: A Machine Learning-Oriented Spiking Neural Networks Library in Python // Frontiers in Neuroinformatics. 2018. Vol. 12. Art. 89.
 14. *Pedregosa F. et al.* Scikit-Learn: Machine Learning in Python // Journal of Machine Learning Research. 2011. Vol. 12. P. 2825–2830.
 15. *Kingma D.P., Ba J.* Adam: A Method for Stochastic Optimization // International Conference on Learning Representations. 2015.
-

CONSTRUCTION OF A “NEURON ACTIVITY–CLASS” MAPPING IN A CONVOLUTIONAL SPIKING NEURAL NETWORK WITH STDP TRAINING

Aleksandr S. Toshchev^[0000-0003-4424-6822]

Kazan (Volga region) Federal University, Kazan, Russia

atoschev@kpfu.ru

Abstract

This paper investigates the problem of constructing a mapping between the spiking activity of a convolutional spiking neural network and the class of an input image. This problem arises after unsupervised training: the convolutional layer forms an event-based representation of the image, but the network itself does not contain a ready-made mechanism for assigning a class label. The aim of the study is to evaluate how informative the spike counters of the convolutional layer are after training with the STDP rule (Spike-Timing-Dependent Plasticity, a biologically plausible learning rule for impulse-based, or spiking, neural networks) and to assess the performance of a linear readout classifier as the number of training presentations increases.

The experiments were conducted on the MNIST dataset, a standard dataset of handwritten digit images. A single-layer architecture was used: one convolutional layer with 32 feature maps, a 5×5 kernel, and an output dimensionality of

$32 \times 24 \times 24$, corresponding to 18432 LIF neurons, where LIF denotes the leaky integrate-and-fire neuron model. The input images were encoded using a deterministic Poisson encoder; the Poisson encoding multiplier was set to 0.006. The presentation duration, that is, the time during which an object was shown to the network, was 100 time steps. Training was performed from scratch using the STDP rule, without resuming from a previously saved state. For evaluation, a protocol was used in which spike counters were collected on 10000 training images and 10000 test images.

In the main experiment, with 15000 STDP training presentations, an accuracy of 0.8883 was achieved. The baseline label-map approach, close to the method of Diehl and Cook, produced an accuracy of about 0.48 under the same configuration. The coverage of the calibrated label map was 219 out of 18,432 neurons, while the average activity was 4,146.9416 spikes per image. The neighboring scale point with 10,000 training presentations yielded an accuracy of 0.8876. The results show that the chosen method for reading out activity has a substantial effect on the final classification quality: a linear readout classifier based on the full vector of network spike counters extracts distributed information that is lost when labels are assigned directly to individual neurons.

Keywords: *spiking neural network, convolutional spiking neural network, readout, spike counters, image classification.*

REFERENCES

1. Gerstner W., Kistler W. Spiking Neuron Models: Single Neurons, Populations, Plasticity. Cambridge: Cambridge University Press, 2002.
2. Maass W. Networks of Spiking Neurons: The Third Generation of Neural Network Models // Neural Networks. 1997. Vol. 10, No. 9. P. 1659–1671.
3. Davies M. et al. Loihi: A Neuromorphic Manycore Processor with On-Chip Learning // IEEE Micro. 2018. Vol. 38, No. 1. P. 82–99.
4. Chertopolokhov V., Devyatkin D., Serov A., Babaev A., Zeleniy A., Kamyshinsky R., Khmelenin D., Gerasimov E., Tsybin I., Trigub A. L., Kashkarov P., Toshchev A., Popov A. Advancing Neural Networks: Innovations and Impacts on Energy Consumption // Advanced Electronic Materials. 2024. Vol. 10, No. 8. Art. 2400029.
5. Toshev A. et al. Critical Analysis of Energy Consumption in Neuro-Computational Systems // IEEE Access. 2026. Art. 60535.

6. *Toshchev A.S.* Generaciya vremennyh signalov iz staticheskikh izobrazhenij dlya podachi na spajkovye nejronnye seti // *Russian Digital Library Journal*. 2026. T. 29. № 3. P. 1061–1077.

7. *Diehl P.U., Cook M.* Unsupervised Learning of Digit Recognition Using Spike-Timing-Dependent Plasticity // *Frontiers in Computational Neuroscience*. 2015. Vol. 9. Art. 99.

8. *Kheradpisheh S.R., Ganjtabesh M., Thorpe S.J., Masquelier T.* STDP-Based Spiking Deep Convolutional Neural Networks for Object Recognition // *Neural Networks*. 2018. Vol. 99. P. 56–67.

9. *LeCun Y., Bottou L., Bengio Y., Haffner P.* Gradient-Based Learning Applied to Document Recognition // *Proceedings of the IEEE*. 1998. Vol. 86, No. 11. P. 2278–2324.

10. *Bi G.Q., Poo M.M.* Synaptic Modifications in Cultured Hippocampal Neurons: Dependence on Spike Timing, Synaptic Strength, and Postsynaptic Cell Type // *The Journal of Neuroscience*. 1998. Vol. 18, No. 24. P. 10464–10472.

11. *Caporale N., Dan Y.* Spike Timing-Dependent Plasticity: A Hebbian Learning Rule // *Annual Review of Neuroscience*. 2008. Vol. 31. P. 25–46.

12. *Izhikevich E.M.* Which Model to Use for Cortical Spiking Neurons? // *IEEE Transactions on Neural Networks*. 2004. Vol. 15, No. 5. P. 1063–1070.

13. *Hazan H. et al.* BindsNET: A Machine Learning-Oriented Spiking Neural Networks Library in Python // *Frontiers in Neuroinformatics*. 2018. Vol. 12. Art. 89.

14. *Pedregosa F. et al.* Scikit-Learn: Machine Learning in Python // *Journal of Machine Learning Research*. 2011. Vol. 12. P. 2825–2830.

15. *Kingma D.P., Ba J.* Adam: A Method for Stochastic Optimization // *International Conference on Learning Representations*. 2015.

СВЕДЕНИЯ ОБ АВТОРЕ



ТОЩЕВ Александр Сергеевич – доцент, к. н., КФУ, Институт информационных технологий и интеллектуальных систем, кафедра программной инженерии, г. Казань.

Aleksandr Sergeevich TOSCHEV – Associate Professor, Ph.D., KFU, Institute of Information Technologies and Intelligent Systems, Department of Software Engineering, Kazan.

email: atoschev@kpfu.ru

ORCID: 0000-0003-4424-6822

Материал поступил в редакцию 16 марта 2026 года