FPGA РЕАЛИЗАЦИЯ ДВУХСЛОЙНОЙ НЕЙРОННОЙ СЕТИ ПРЯМОГО РАСПРОСТРАНЕНИЯ ДЛЯ РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЙ

Субботенко О. Р., Вашкевич М. И.
Кафедра электронных вычислительных средств,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: subbotenkoolga2004@gmail.com, vashkevich@bsuir.by

В работе представлена аппаратная реализация двухслойной нейронной сети для распознавания рукописных цифр с использованием ПЛИС типа FPGA. Также проанализировано влияние точности представления параметров НС на её производительность и аппаратные затраты ПЛИС.

Введение

В настоящее время нейронные сети (НС) играют важную роль в задачах компьютерного зрения и обработки естественного языка. Зачастую для обучения и применения НС используют графические процессоры (GPU), так как они обеспечивают параллельную обработку данных. К недостаткам можно отнести высокое энергопотребление и невозможность выбора формата чисел [1].

Альтернативным решением является реализация НС на базе программируемых логических интегральных схемы (ПЛИС) типа FPGA [1-2]. К их преимуществам перед GPU можно отнести возможность выбора точности представления параметров, большая производительность и меньшее энергопотребление.

В работе представлена реализация двухслойной нейронной сети на ПЛИС для распознавания рукописных цифр из базы MNIST с использованием арифметики с фиксированной запятой. Проанализирована зависимость между разрядностью коэффициентов НС, аппаратными затратами и точностью распознавания.

I. Разработка двухслойной нейронной сети

Предлагаемая в работе НС для классификации изображений рукописных цифр имеет два полносвязных слоя. На рис. 1 приведена ее структура.

Для обучения и тестирования НС была использована база MNIST, содержащая изображения размером 28×28 пикселей рукописных цифр от 0 до 9 в оттенках серого. Она состоит из 70 тыс. изображений, из которых 60 тыс. относятся к обучающей выборке, а 10 тыс. – к тестовой. На вход НС подаются изображения, предварительно преобразованные в вектор.

Скрытый слой принимает входные данные и рассчитывает результат согласно выражению:

$$z_t^{[1]} = \sum_{s=0}^{783}(w_{t,s}^{[1]} \cdot x_s) + b_t^{[1]},$$

где $z_t^{[1]}$ – значения преактиваций скрытого слоя, $t \in [0;13]$; $w_{t,s}^{[1]}$ – веса 1-го слоя; x_s – входные значения (пиксели); $b_t^{[1]}$ – смещение t-го нейрона 1-го слоя.

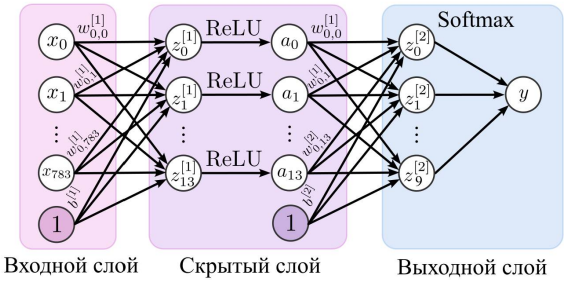

Рис. 1 – Структура двухслойной нейронной сети

Результаты вычислений поступают на вход активационной функции ReLU, которая описывается выражением:

$$f(z) = \mathrm{ReLU}(z) = \max(z,0).$$

Далее эти значения подаются на вход выходного слоя, где преобразуются в выходные по формуле:

$$z_t^{[2]} = \sum_{s=0}^{13}(w_{t,s}^{[2]} \cdot a_s) + b_t^{[2]},$$

где $z_t^{[2]}$ – значения преактиваций выходного слоя, $t \in [0;9]$; $w_{t,s}^{[2]}$ – веса 2-го слоя; a_s – выходы скрытого слоя; $b_t^{[2]}$– смещение t-го нейрона 2-го слоя.

Вычисленные 10 значений поступают на вход функции softmax:

$$\mathrm{softmax}(z_i) = \frac{\exp(z_i)}{\sum_{j=1}^{n}\exp(z_j)},$$

где z_i – элемент входного вектора.

Обучение НС выполнялось с использованием языка Python и библиотеки PyTorch. В качестве функции потерь использовалась перекрестная энтропия, оптимизация выполнялась методом Adam со скоростью обучения $\eta = 0{,}0003$ и коэффициентом L2-регуляризации $\lambda = 0{,}0001$. Обучение проходило в 200 эпох, благодаря чему модели

Работа выполнена в совместной учебной лаборатории БГУИР-YADRO
https://www.bsuir.by/ru/kaf-informatiki/yadro

удалось достигнуть высокой точности и избежать переобучения.

II. Реализация НС на FPGA

Для аппаратной реализации двухслойной нейронной сети была выбрана отладочная плата PYNQ Z2 на базе ПЛИС Zynq-7000. Zynq представляет собой систему на кристалле (СнК), объединяющую процессор ARM Cortex-A9 и программируемую логику FPGA. Для упрощения разработки и тестирования на данной платформе используется PYNQ (Python Productivity for Zynq). PYNQ – это специализированный дистрибутив Linux для плат на основе Xilinx Zynq и набор библиотек Python (pynq), которые предоставляют API для взаимодействия с программируемой логикой процессорной системой через IP-ядра, такие как DMA, GPIO и др.

IP-блок двухслойной НС описан на языке SystemVerilog. Полносвязные слои реализуются на базе MAC-ядер: 14 ядер находятся в скрытом слое и 10 в выходном. Каждое MAC-ядро производит операцию умножения входных значений на веса слоя, хранящиеся в блочной памяти FPGA. Функция ReLU реализуется при помощи мультиплексора, управляемого знаковым разрядом аргумента. Вместо softmax используется argmax, так как он требует меньших затрат. В блоке argmax осуществляется поиск индекса наибольшего элемента выходного вектора второго слоя. Тактовая частота IP-ядра составила 40 МГц.

Изображение из процессорной системы передается в блок IP-ядро НС по интерфейсу AXILite. По окончании вычислений результат распознавания передаётся обратно в процессорную систему по тому же интерфейсу (рис. 4).

III. Тестирование НС

Исследовалось влияние представления весов НС на ее точность и аппаратные затраты FPGA. Разрядность дробной части весов изменялась от 2 до 12. В каждом случае тестирование производилось на всех тестовых изображениях из базы MNSIT. Для анализа полученных результатов строились матрицы спутывания, одна из которых приведена на рис. 2.

С увеличением разрядности весов точность предсказаний растет. В то же время анализ аппаратных затрат FPGA показывает, что количество требуемых LUT (Look-Up Table) и FF (Flip-Flop) также растет(рис. 3).

По полученным результатам можно сделать вывод, что наиболее оптимальным является представление коэффициентов с 8 разрядами в дробной части. В этом случае точность достигает довольно высокого значения (94,24%), а аппаратные затраты принимают приемлемые значения.

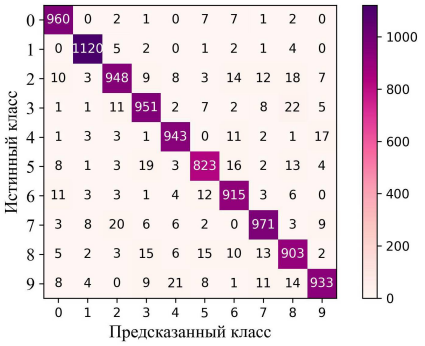

Рис. 2 – Матрица спутывания для коэффициентов НС с 10-разрядным представлением дробной части

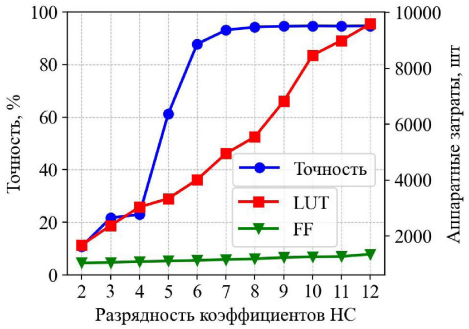

Рис. 3 – Точность и аппаратные затраты НС

IV. Заключение

В работе описан способ реализации двухслойной НС для распознавания рукописных цифр на базе платформы PYNQ Z2. Также была исследована зависимость между форматом представления весов НС и аппаратными затратами FPGA.

1. Sayed R., Draz H. H., Azmi H. HW-SW co-design of image classification accelerator on FPGA // The Journal of Supercomputing. – 2025. – Т. 81. – №. 8. – P. 1-26.
2. Кривальцевич Е. А., Вашкевич М. И. Исследование аппаратной реализации нейронной сети прямого распространения для распознавания рукописных цифр на базе FPGA // Доклады БГУИР. – 2025. – Т. 23. – №. 2. – P. 101-108.

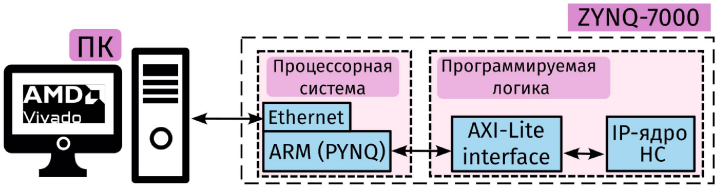

Рис. 4 – Реализация НС для распознавания рукописных цифр на базе платформы PYNQ Z2