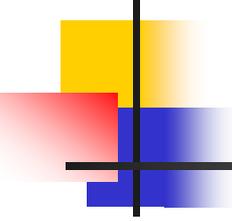


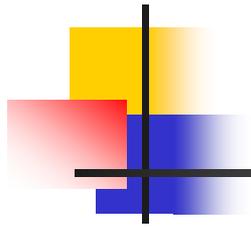
Нейронные сети и нейрокомпьютеры

Буряк Дмитрий,
к.ф-м.н

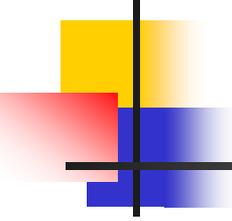


План

1. История появления
2. Структура искусственной нейронной сети
3. Обзор нейросетевых архитектур
4. Примеры применения НС
5. Реализация НС на GPU
6. Нейрокомпьютеры



ИСТОРИЯ ПОЯВЛЕНИЯ

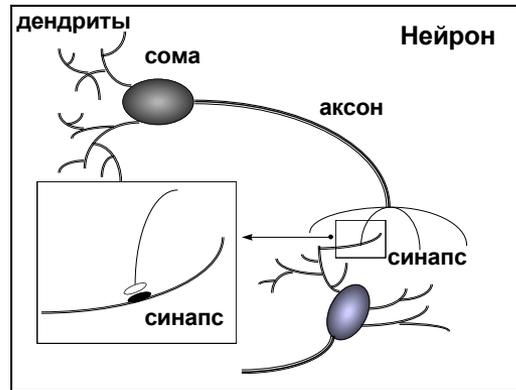


Почему нейронные сети (НС)

- Обучение на примерах
- Массовый параллелизм
- Устойчивость к зашумленным данным
- Адаптивность

Появление искусственных НС

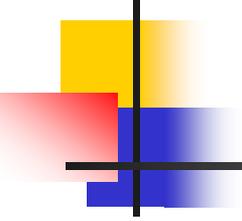
□ Исследования головного мозга



- общее число нейронов 10^{10}
- число связей 10^{14}
- взаимодействие нейронов
- посредством электрических
- импульсов

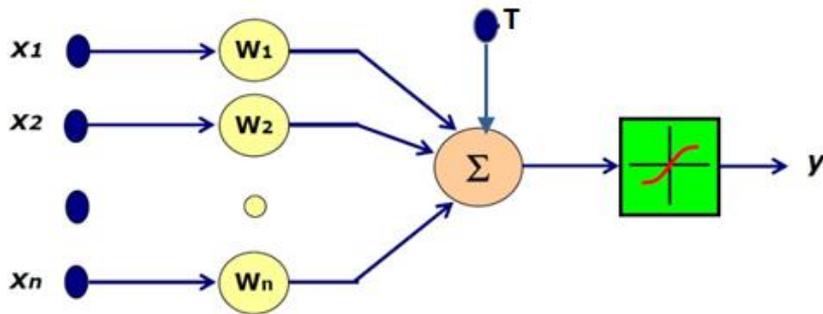
□ История появления НС

- 1943г. У. МакКаллок, У. Питтс. Статья о вычислениях в сетях формальных нейронов.
- 1951г. М. Минский. Первый экспериментальный нейрокомпьютер Snark
- 1961г. Ф. Розенблат. Создание перцептрона, идея обучения на примерах.
- 1969г. М. Минский, С. Пейперт. Книга «Перцептроны».
- 1974г. П.Дж.Вербос, А.И.Галушкин. Алгоритм обратного распространения ошибки.
- 1986г. Д.И.Румельхарт, С.И.Барцев. Развитие метода обратного распространения ошибки
- 2000-е гг. Алгоритмы глубокого обучения.



СТРУКТУРА ИСКУССТВЕННОЙ НС

Искусственный нейрон



- веса W_1, W_2, \dots, W_n ;
- порог (смещение) T ;
- функция активации f ;

- выход нейрона: $y = f\left(\sum_{i=1}^n w_i x_i + T\right)$

Виды функции активации

- пороговая

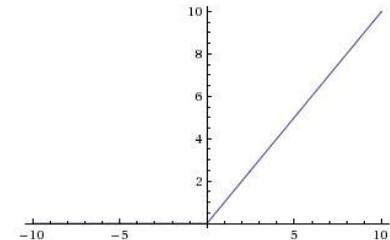
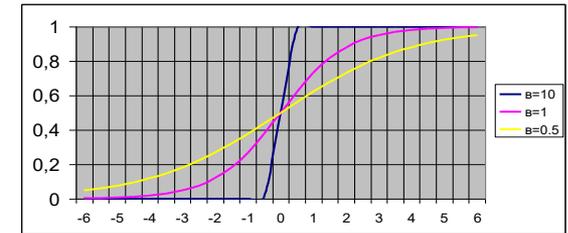
$$y = f(u) = \begin{cases} 1, u > 0 \\ 0, u \leq 0 \end{cases}$$

- сигмоидальная

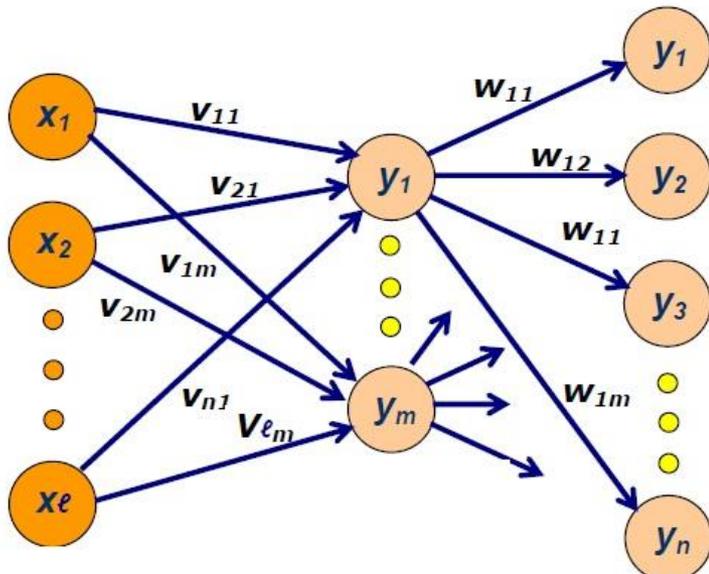
$$f(u) = \frac{1}{1 + e^{-\beta u}} \quad f(u) = th(\beta u) = \frac{e^{\beta u} - e^{-\beta u}}{e^{\beta u} + e^{-\beta u}}$$

- ReLU – Rectified Linear Unit

$$f(u) = \max(0, u)$$



Сеть нейронов



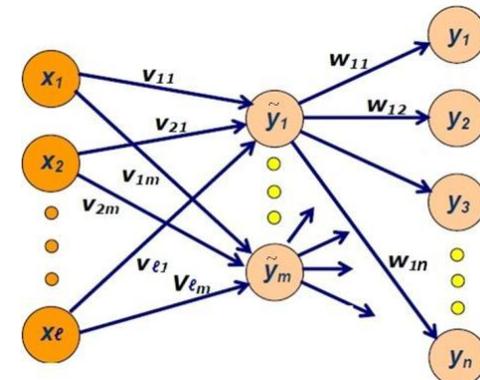
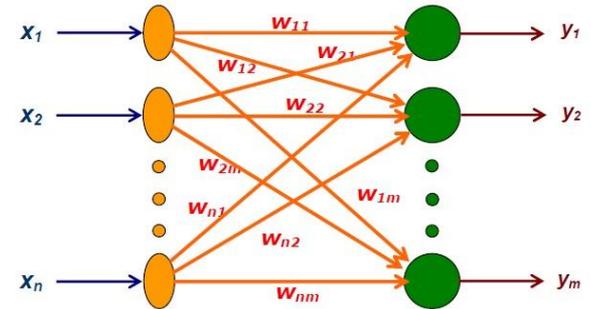
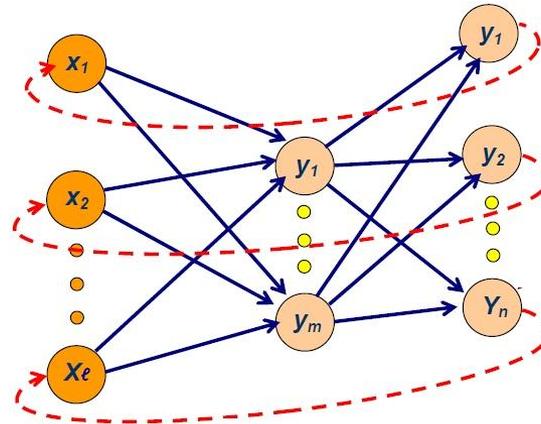
- Искусственная НС – направленный граф;
- Вершины – нейроны (входные, выходные);
- Дуги – синаптические связи с весами;
- Входные данные – входной вектор;
- Выходные данные – выходной вектор;
- Значения нейронов передаются по связям;
- Значения весов определяются через обучение

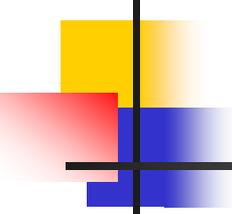
Виды архитектур НС

□ Сети прямого распространения

- однослойный персептрон;
- многослойный персептрон

□ Рекуррентные сети





Создание НС

□ Выбор архитектуры

- тип архитектуры
- структура нейрона
- размер входного и выходного слоев;
- количество слоев и нейронов в слое
- связи между нейронами

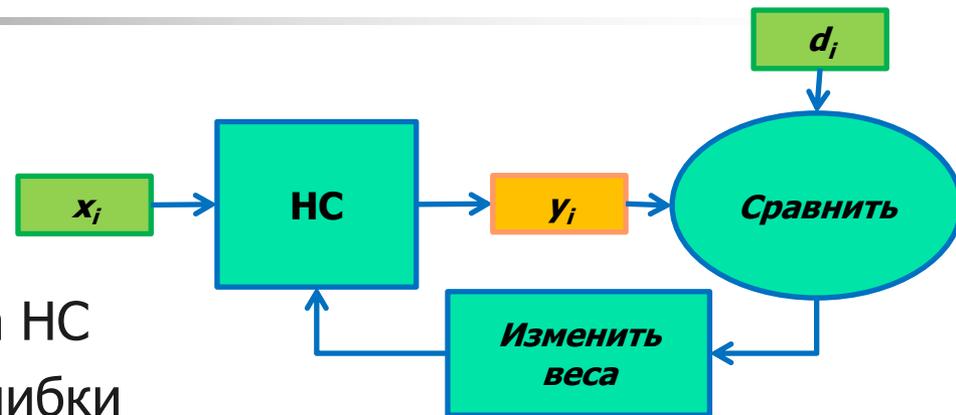
□ Определение значений весов - обучение

- обучающая выборка
- алгоритм обучения
- инициализация весов
- проведение обучения

Обучение

□ Итерационный процесс

- выбор примера обучающей выборки
- вычисление значения выхода НС
- оценка значения функции ошибки
- коррекция весов НС

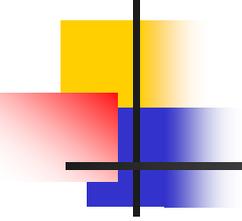


□ Виды алгоритмов обучения

- С учителем: минимизация между реальными выходами и ожидаемыми.
- Без учителя: конкуренция нейронов между собой.
- С подкреплением: имеется информация о правильности выхода.

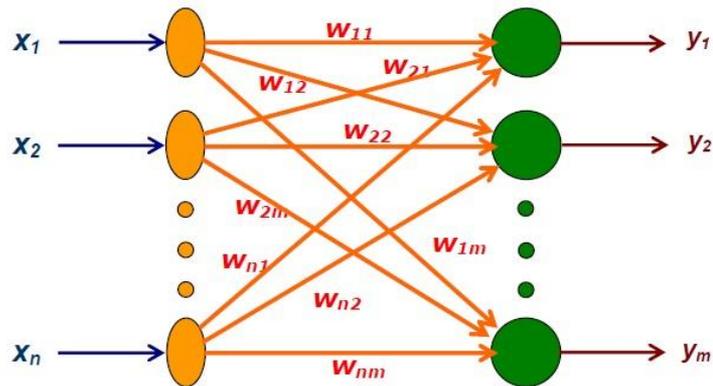
□ Пример эффективности обучения

- Задача: оптимальное управление при ограниченных ресурсах.
- Степень обучения: низкая, средняя, высокая, различная.



ОБЗОР НЕЙРОСЕТЕВЫХ АРХИТЕКТУР

Однослойный персептрон

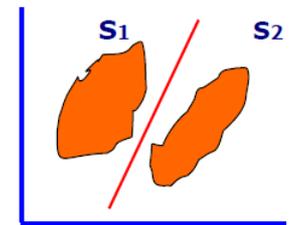


- Входной слой, выходной слой
- Функция активации: сигмоида, $th(u)$
- Обучение с учителем
- Пример алгоритма обучения: градиентный спуск

$$E(w) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^m (y_j^i - d_j^i)^2$$

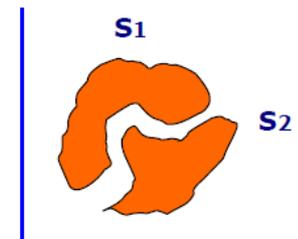
□ Линейная разделимость

- Вход – 2 нейрона, выход – 1 нейрон → НС – прямая на плоскости
- Классификация линейно разделимых классов

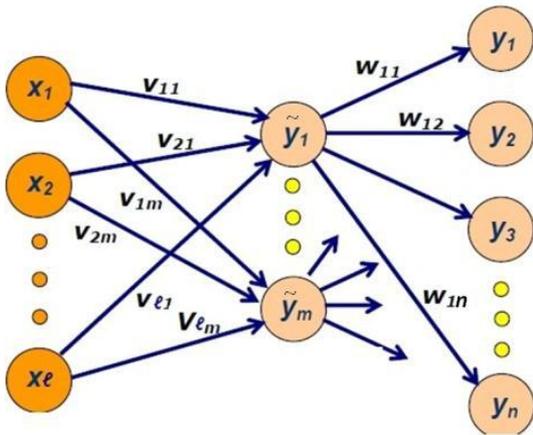


□ Свойства

- «+»: Скорость, память
- «-»: Ограниченность решаемых задач



Многослойный персептрон



- Входной слой, внутренние слои, выходной слой
- Функция активации: сигмоида, $th(u)$
- Обучение с учителем
- Основной алгоритм обучения: обратное распространение ошибки

$$E(w) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^m (y_j^i - d_j^i)^2$$

$$\Delta w_{pq}^k = -\eta \delta_q^k f_p^{k-1}$$

- Размер обучающей выборки $> O(K)$.

□ Свойства

- «+»: универсальность, эффективность;
- «+»: большой выбор алгоритмов обучения;
- «-»: проблемы обучения.

$$\delta_p^{k-1} = \frac{df_p^{k-1}}{du_p^{k-1}} \left[\sum_q \delta_q^k w_{pq}^k \right]$$

Алгоритмы обучения

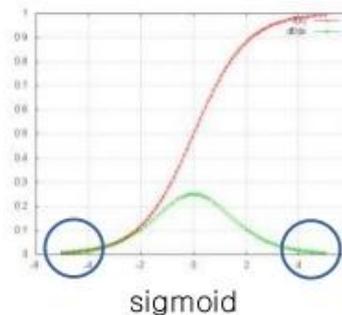
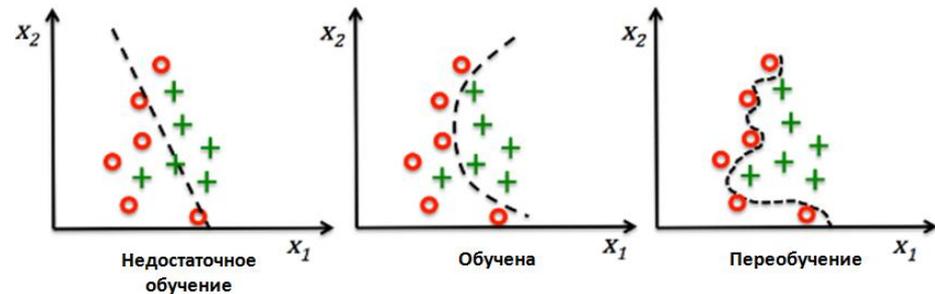
□ Минимизация функции ошибки

- 1-го порядка: градиент функции
- 2-го порядка: матрица вторых производных.
- Эвристические
- Стохастические

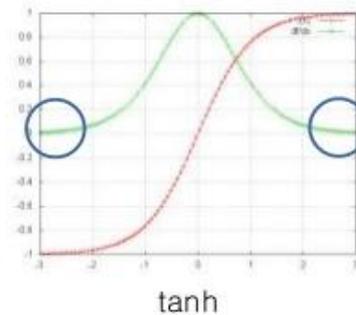
$$E(w) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^m (y_j^i - d_j^i)^2$$

□ Проблемы обучения

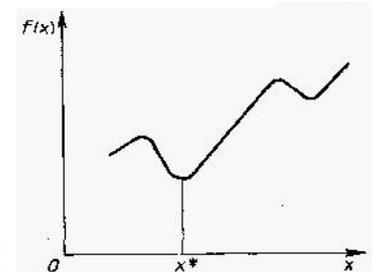
- переобучение
- локальные минимумы функции ошибки



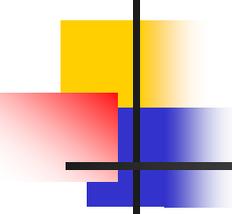
sigmoid



tanh



- близость градиента к 0



Универсальный аппроксиматор

- Многослойный персептрон - аппроксиматор функции $F: R^{m_0} \rightarrow R$
- Каково минимальное количество скрытых слоев для аппроксимации любой функции?
- Теорема существования НС из 2 слоев

$$F(x_1, \dots, x_{m_0}) = \sum_{i=1}^{m_1} \alpha_i \varphi \left(\sum_{j=1}^{m_0} w_{ij} x_j + b_i \right)$$

- Входной слой: m_0
Внутренний слой: m_1
Выходной слой: 1 линейный нейрон
- Один внутренний слой – не всегда оптимальное решение

Глубокое обучение (Deep learning)

❑ Общие сведения

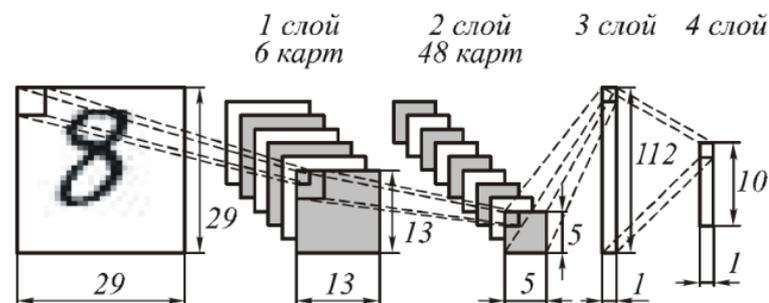
- тип алгоритмов машинного обучения, которые пытаются моделировать высокоуровневые абстракции в данных, используя архитектуры, состоящие из множества нелинейных трансформаций;
- большое число слоев и нейронов -> большое число весов;
- высокие требования к вычислительным ресурсам.

❑ Типы архитектур

- сверточные нейронные сети
- глубокие сети доверия

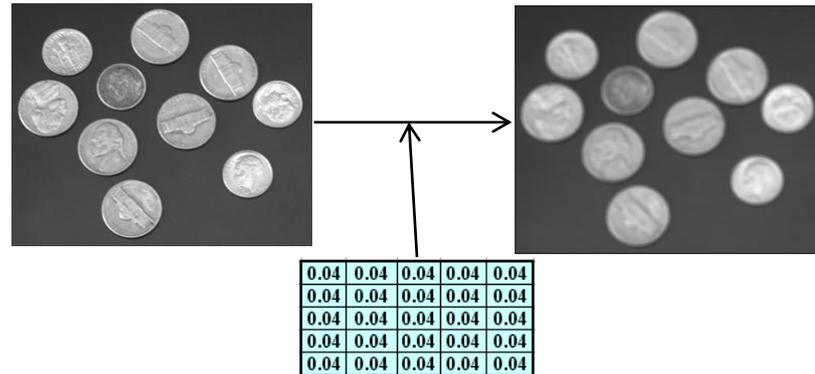
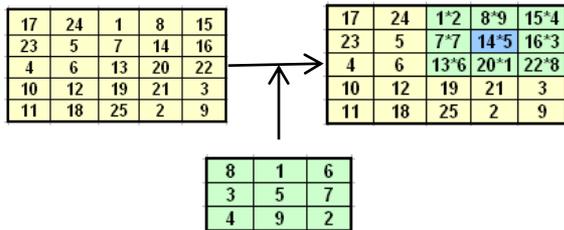
❑ Решаемые задачи

- компьютерное зрение
- распознавание речи

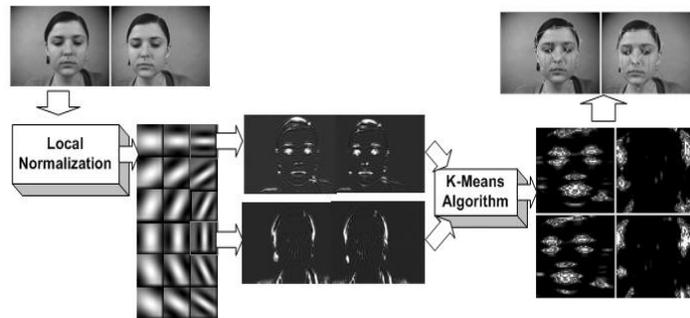


Сверточные нейронные сети (1)

❑ Операция свертки



❑ Решение задачи обнаружения лица (Vazanov et al, 2004)

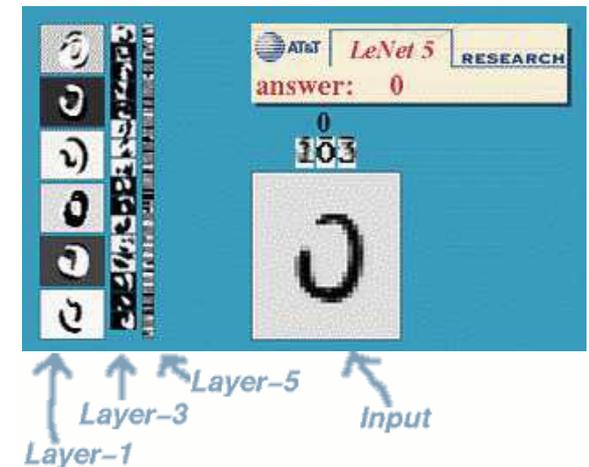
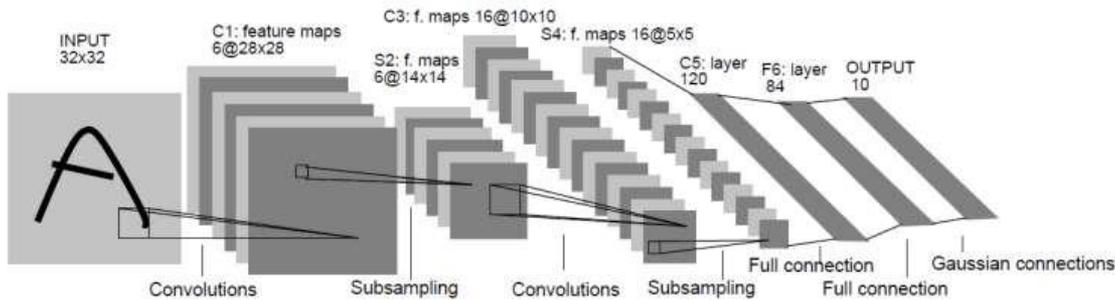


❑ Сверточные НС – поиск оптимального набора операций свертки

Сверточные нейронные сети (2)

□ НС свертки (LeCun, 1998)

- Обработка локальных областей
- Совместное использование весов
- Устойчивость к искажениям и шуму

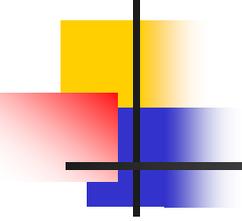


□ Обучение

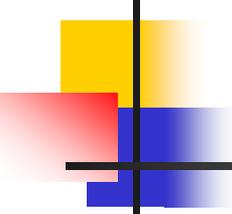
- Случайный градиентный спуск (Stochastic Gradient Descend)
- Регуляризация

$$E(w) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^m (y_j^i - d_j^i)^2 + C \|w\|^2$$

- Ошибка распознавания на базе MNIST 0.7%



ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ



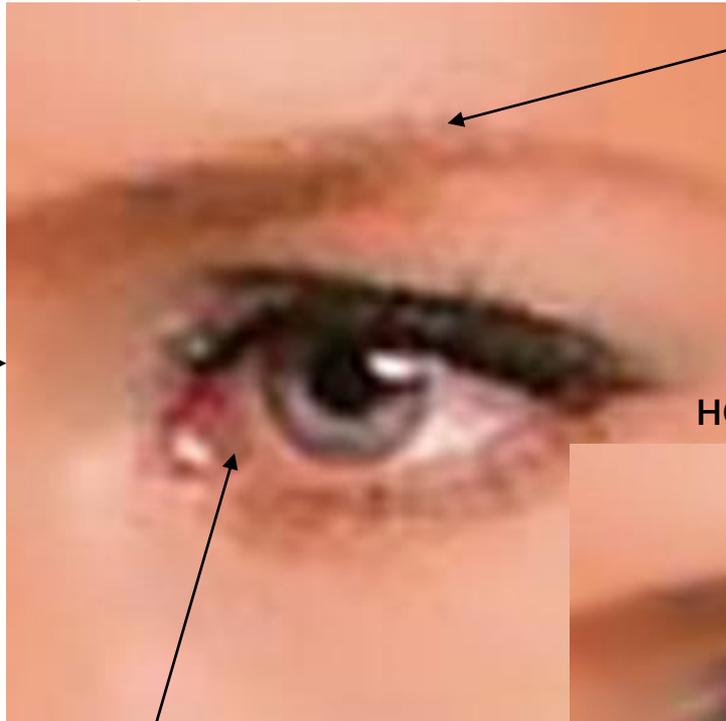
Области применения ИС

- Классификация образов
- Кластеризация
- Аппроксимация функций
- Прогнозирование
- Оптимизация
- Ассоциативная память
- Управление

НС для масштабирования изображений



бикубическая интерполяция



шумы



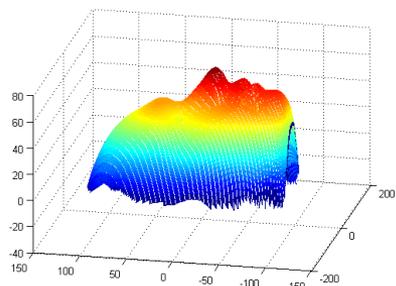
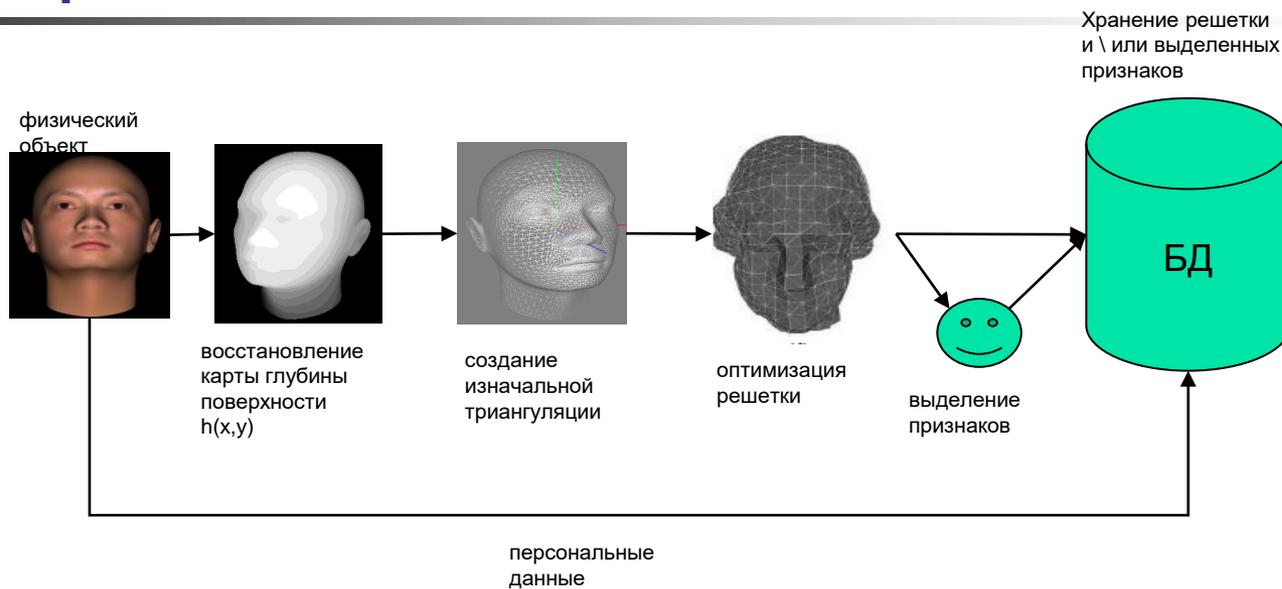
нейронная сеть



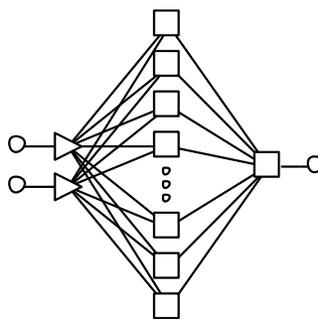
артефакты



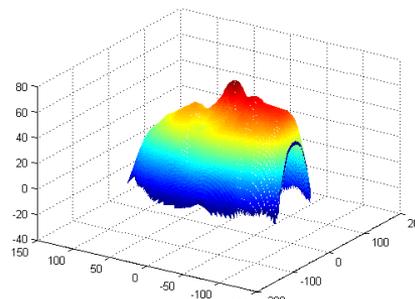
3D распознавание лиц



оригинальная карта высот
9852x4 байт
(значений точек)



128 нейронов в скрытом слое



нейросетевая аппроксимация,
погрешность 2%

513x4 байт
(значений весовых коэф.)

Классификация изображений

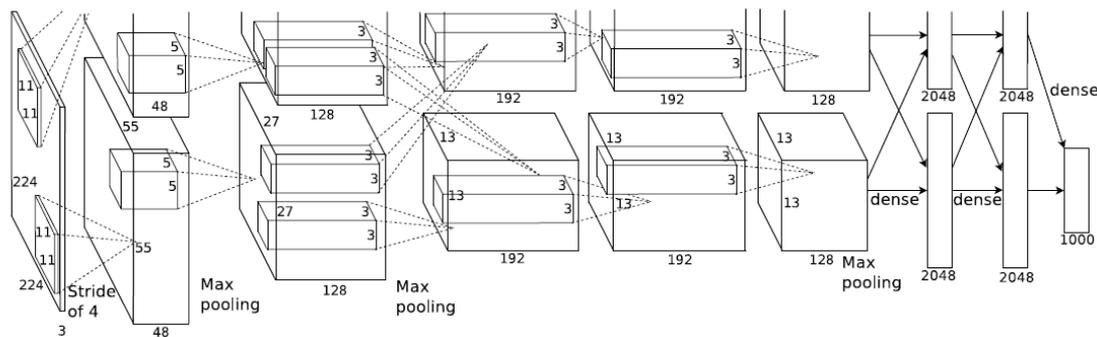
ImageNet

- 1.2M изображений (обучающая выборка)
- 1000 классов



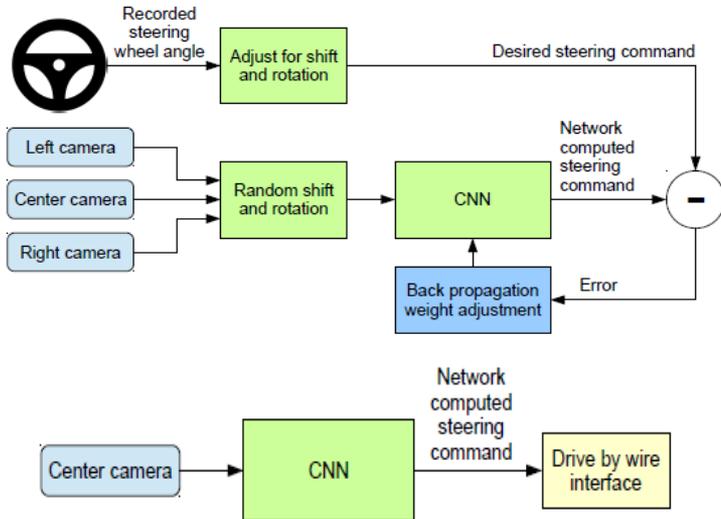
Alex Krizhevsky et al, 2012

- Более 550000 нейронов
- Ошибка классификации:
Top-5 15.3%

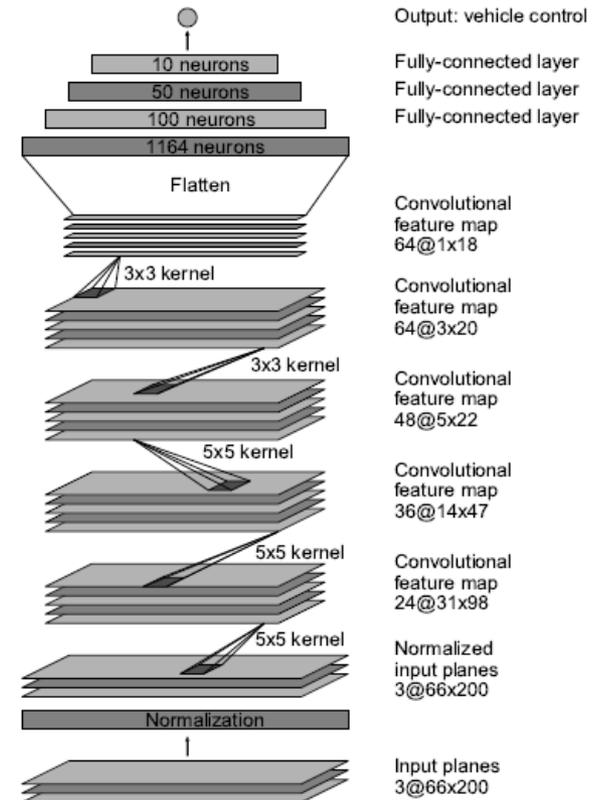


Беспилотный автомобиль

☐ Dave-2 (NVIDIA)

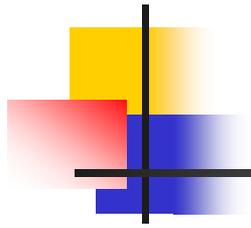


- Обучение на действиях водителя
- Управление на основе данных с одной камеры



☐ Архитектура HC

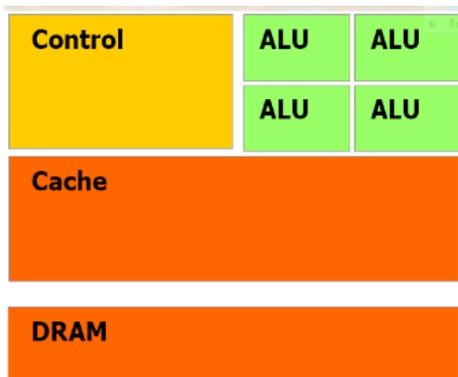
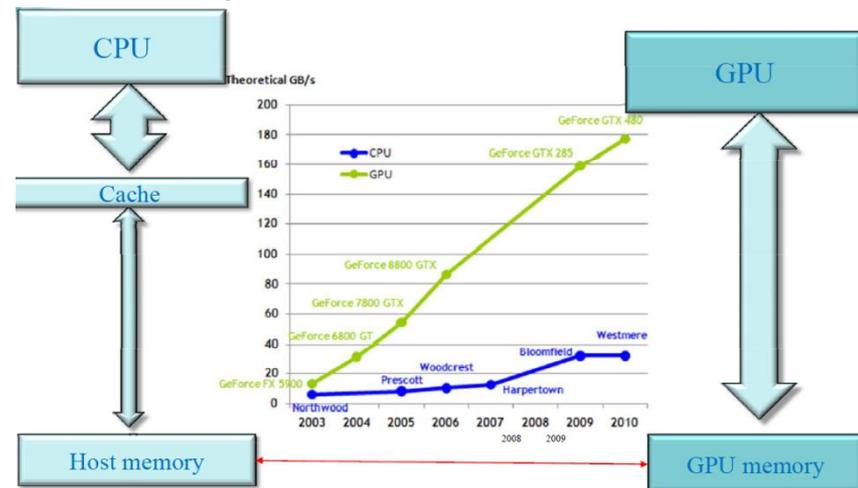
- 27М связей
- 250К параметров



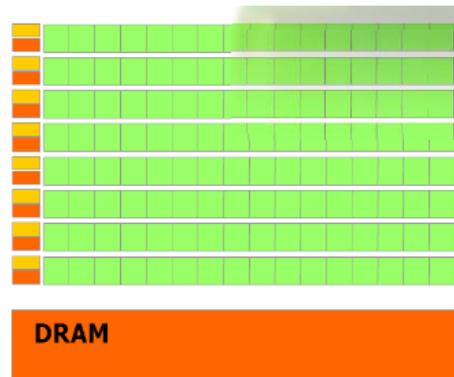
РЕАЛИЗАЦИЯ НА GPU

Graphics processing units (GPU)

- Большое число вычислительных ядер, выполняющихся параллельно.
- Разработано для обработки графики
- Применимо для многих вычислительных задач
- Доступность



CPU



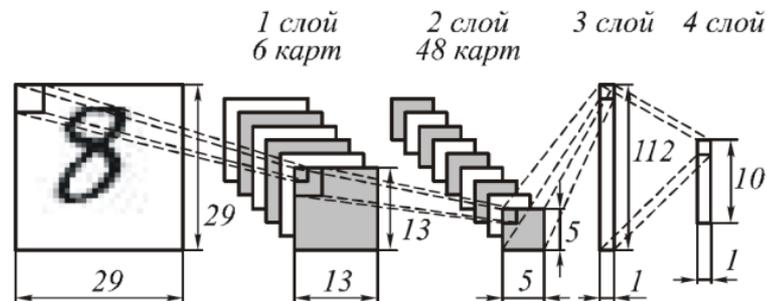
GPU

- CUDA (compute unified device architecture) – C + расширение для GPU
- CUDA: упрощение доступа к памяти
- CUDA: блоки потоков, параллельное выполнение

CUDA: распознавание цифр

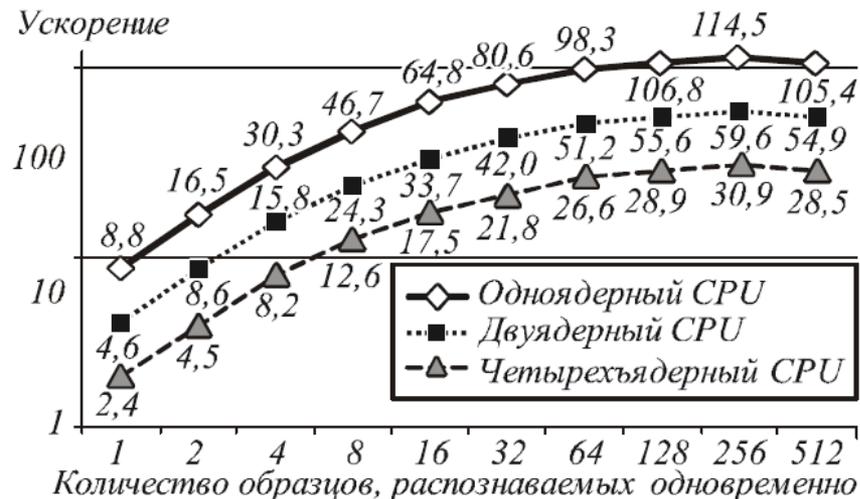
□ Задача:

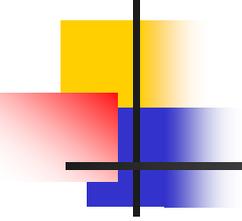
- Распознавание рукописных цифр
- Сверточная НС
- Обучение: обратное распространение ошибки: 60000 раз по одному вектору из 500



□ Результаты

- ускорение обучения 5.96 раз (одноядерный CPU)

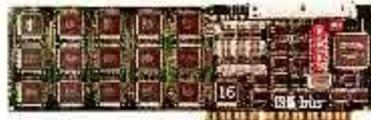




НЕЙРОКОМПЬЮТЕРЫ

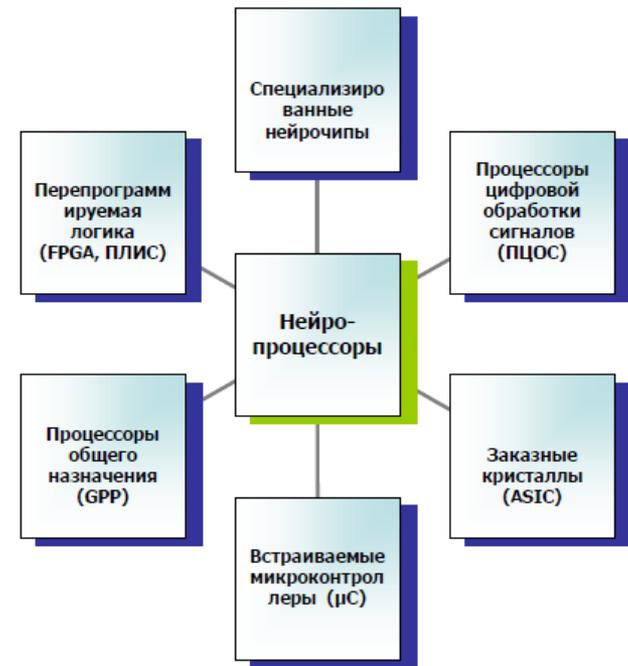
Нейрокомпьютер

- Обработка информации на основе принципов работы естественных НС
- Высока степень параллелизма
- Устойчивость к помехам и повреждениям
- Использование ненадежных элементов



Примеры

- Нейрокомпьютер Synapse (Siemens);
- Нейрокомпьютер «Электронный мозг» (США, для обработки аэрокосмических изображений)



Мемристоры

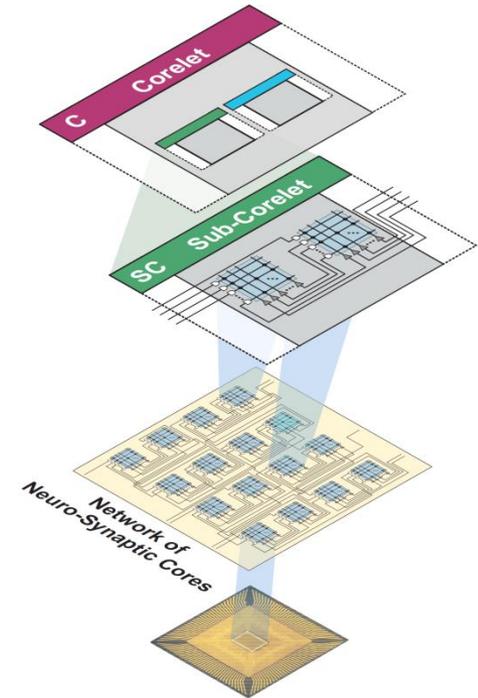
- Мемристор - пассивный элемент в микроэлектронике, способный изменять свое сопротивление в зависимости от протекавшего через него заряда.
- Теория: Леон Чуа, 1971г.
- Первый образец: Р. Стэнли Уильямс, HP, 2008г.
- Материалы: твердотельные (оксид металлов и кремния), органические
- Производители: HP, Hynix, НИЦ «Курчатовский институт», ТюмГУ и др.



- нейристоры – функциональные аналоги нейронов на основе схем из мемристоров и конденсаторов, HP 2012г.

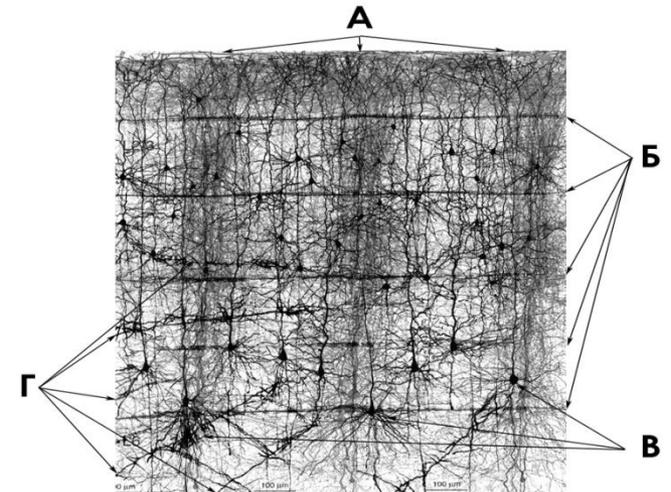
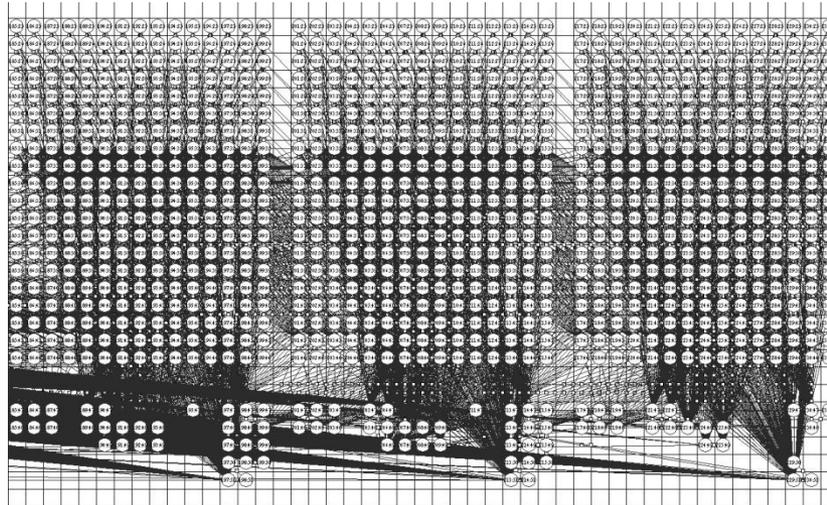
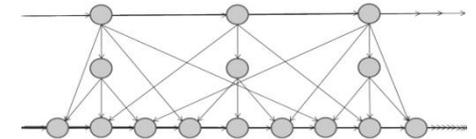
Проект C2S2 (IBM)

- Blue Brain Project, IBM, 2005
- Моделирование нейронов коры мыши, IBM Almaden Research center, 2007
- C2S2 (Cognitive Computing via Synaptronics and Supercomputing), SyNAPSE IBM, DARPA, 2008
- Corelet
- Архитектура фон Неймана -> КОГНИТИВНЫЙ КОМПЬЮТЕР



Искусственные нейрокогнитивные системы (ТюмГУ, ТАСО)

- Нейрон ТАСО-2005
- Активация дополнительных нейронов при обучении
- Моделирование неокортикальных колонок



- Обработка семантики текста на естественном языке
- Управление газопроводом