

## ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ В ДИСКРЕТНОЙ МАТЕМАТИКЕ

УДК 007.52

### АРХИТЕКТУРА НЕЙРОННОЙ СЕТИ С ПОПАРНО ПОСЛЕДОВАТЕЛЬНЫМ РАЗДЕЛЕНИЕМ ОБРАЗОВ

П. Ш. Гейдаров

*Институт систем управления НАН Азербайджана, г. Баку, Азербайджанская Республика*

На основе архитектуры нейронной сети, реализующей метод ближайшего соседа, рассматривается архитектура нейронной сети с попарно последовательным разделением образов без использования аналитических выражений и набора выбранных эталонов. Изучаются возможности данной архитектуры. Показано, что такая архитектура может быть применена к задаче распознавания с очень большим количеством образов. Предлагаемая нейронная сеть отличается простой и понятной архитектурой, возможностью простого обучения нейронной сети с добавлением в неё новых распознаваемых образов без необходимости изменения предыдущих настроек сети.

**Ключевые слова:** архитектуры нейронных сетей, определяемые нейронные сети, нейрокompьютер, сверточные сети, алгоритмы обучения нейронных сетей.

DOI 10.17223/20710410/41/10

### THE ARCHITECTURE OF A NEURAL NETWORK WITH A SEQUENTIAL DIVISION OF IMAGES INTO PAIRS

P. Sh. Geidarov

*Institute of Control Systems of ANAS, Baku, Republic of Azerbaijan***E-mail:** plbaku2010@gmail.com

The known classical architectures of neural networks have many weak properties and high limitations such as great difficulties in choosing proper parameters (the numbers of neurons, connections, layers), in learning and in expanding a learned network and some others. In this paper, to overcome these shortcomings, we consider the architecture of a neural network in which the recognition is performed in pairs. The neural network is created on the basis of the neural network architecture that implements the method of the nearest neighbor, without using analytical expressions and a set of selected samples. The paper shows that such an architecture can be applied to the recognition problem with a very large number of images (classes). At the same time, the proposed neural network has a simple architecture, with the possibility of simpler learning of the neural network. The neural network architecture allows to add new recognizable images to the neural network without changing the previous network settings.

**Keywords:** neural network architectures, neural networks, neurocomputer, convolutional networks, neural network training algorithms.

## Введение

Создание простых и понятных архитектур нейронных сетей с прозрачной архитектурой, способных решать сложные задачи, по-прежнему является актуальной задачей. Известные в настоящее время классические архитектуры нейронных сетей [1–3] имеют ряд слабых сторон и ограничений, среди которых такие, как:

1. Сложность выбора структуры архитектуры нейронной сети и её параметров (количества нейронов, связей, слоёв). Эти параметры выбираются либо эмпирически, либо при помощи приближённых подходов и рекомендаций, либо имеют весьма сложную многослойную форму, как, например, для сетей с глубоким обучением [4–6].
2. Сложность обучения нейронной сети с большим количеством распознаваемых образов. Чем больше количество распознаваемых образов, тем сложнее обучаются нейронные сети и тем хуже, как правило, результат обучения. Это приводит к ограничению возможного количества образов, используемых в задачах распознавания с применением нейронных сетей.
3. Сложность расширения уже обученной нейронной сети и добавления к ней новых образов, которые часто требуют переобучения сети с полным или частичным изменением предыдущих весовых и пороговых значений.

В данной работе рассматривается возможность создания прозрачной архитектуры нейронной сети на основе нейронной сети, реализующей метрические методы распознавания, что позволит обойти приведённые трудности.

В работах [7, 8] рассмотрены архитектуры нейронных сетей, реализованные на основе метрических методов распознавания (НСММР) [9]. В [10] выполнено преобразование этих сетей в многослойный персептрон с полной системой связи, которое показало, что данные сети являются частным случаем многослойного персептрона. Но при этом в отличие от классических схем многослойного персептрона структура и параметры НСММР, такие, как количество нейронов, связей, слоёв, а также значения весов и порогов вычисляются аналитически, исходя из начальных условий задачи — количества распознаваемых образов, эталонов, используемых метрических выражений. В частности, на рис. 1 приведена схема трёхслойной нейронной сети прямого распространения с пороговой функцией активации, реализующей метод ближайшего соседа [7, 10].

В качестве соседних элементов метода ближайшего соседа используется набор отобранных эталонов из обучающей выборки. Набор эталонов может отбираться как интуитивно, так и при помощи алгоритма обучающего отбора [7].

Структура данной сети (количество нейронов, слоёв и связей) строго определяется согласно схеме на рис. 2. Каждый нейрон первого слоя выполняет попарное сравнение изображений двух эталонов. При этом значения весов нейронов первого слоя определяются аналитически на основе метрических выражений, например для рис. 2, б по выражению

$$w_{c,r}^{(1)} = d_1^2 - d_2^2 = ((c_1 - c_p)^2 + (r_1 - r_p)^2) - ((c_2 - c_p)^2 + (r_2 - r_p)^2), \quad (1)$$

где  $(c_1, r_1)$  и  $(c_2, r_2)$  являются координатами точек (ячеек таблицы весов) до ближайшей точки (ячейки) изображения эталона с координатами  $(c_p, r_p)$  (рис. 2, а). В качестве выражений меры близости могут использоваться и другие более простые или сложные выражения [9], отличные от (1).

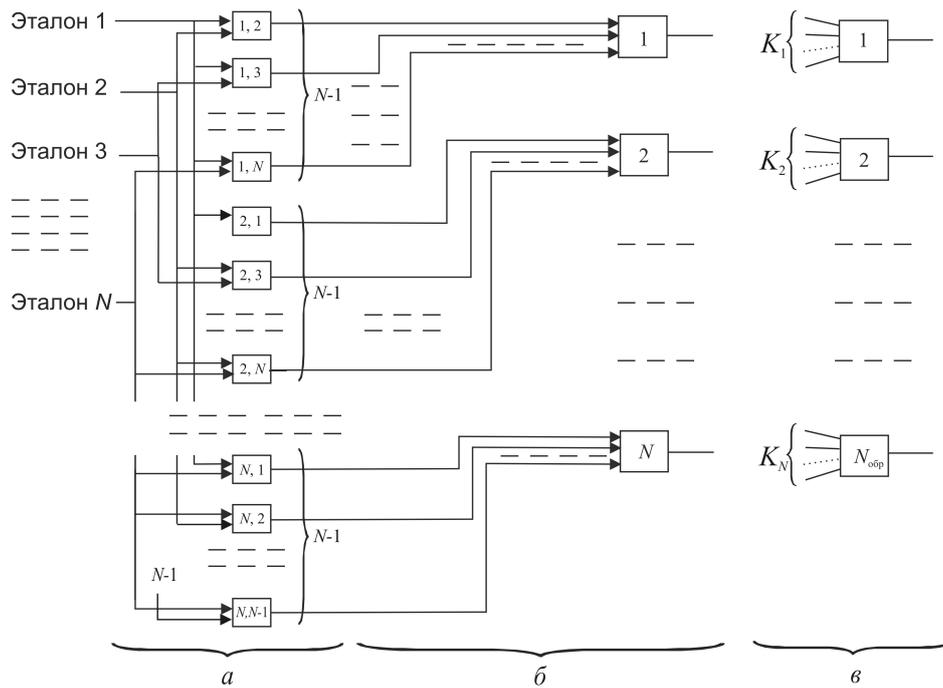


Рис. 1. Архитектура сети на основе метода ближайшего соседа для  $N$  эталонов и  $N_{обр}$  образов (классов)

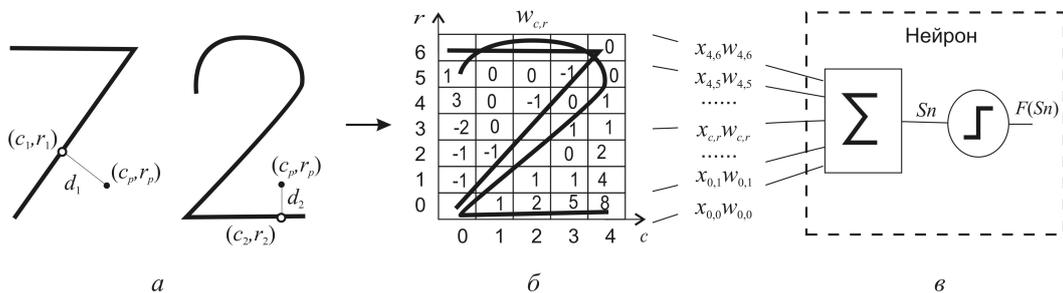


Рис. 2. Расстояния  $d_1$  и  $d_2$  для точки  $(c_p, r_p)$  (а); таблица весов для эталонов «2» и «7» (б); нейрон с пороговой функцией активации (в)

Функции состояния  $Sn_{i,j}^{(1)}$  и активации  $f(Sn_{i,j}^{(1)})$  для каждого нейрона первого слоя определяются по формулам

$$Sn_{i,j}^{(1)} = \sum_{r=0}^R \sum_{c=0}^C x_{c,r} w_{c,r}^{(1)}, \quad f(Sn_{i,j}^{(1)}) = \begin{cases} 1, & \text{если } Sn_{i,j}^{(1)} < 0, \\ 0, & \text{если } Sn_{i,j}^{(1)} > 0, \end{cases}$$

где  $C, R$  — количество столбцов и строк в таблице весов (на рис. 2, б  $C = 5, R = 7$ ). Значения весов для каждого входа нейронов второго и третьего слоя равны 1:

$$w_{i,j}^{(2)} = w_{i,j}^{(3)} = 1.$$

Количество нейронов второго слоя равно количеству используемых эталонов  $n_2 = N$ . Значения функций состояния и активации для  $k$ -го нейрона второго слоя определяются по выражениям

$$Sn_k^{(2)} = \sum_{j=1, j \neq k}^N f \left( Sn_{k,j}^{(1)} \right); \quad (2)$$

$$f \left( Sn_k^{(2)} \right) = \begin{cases} 1, & \text{если } Sn_k^{(2)} \geq (N - 1) = B^{(2)}, \\ 0, & \text{если } Sn_k^{(2)} < (N - 1) = B^{(2)}. \end{cases} \quad (3)$$

Здесь  $B^{(2)} = N - 1$  — пороговое значение нейрона второго слоя. Если выход  $k$ -го нейрона второго слоя равен  $y_k^{(2)} = 1$ , то это означает, что объекту  $\bar{x}$  на входе нейронной сети соответствует  $k$ -й эталон. Нейроны третьего слоя объединяют выходы эталонов одного  $k$ -го образа в единый выход  $y_k^{(3)}$ . Функции состояния  $Sn_k^{(3)}$  и активации  $f \left( Sn_k^{(3)} \right)$  для нейрона третьего слоя определяются по формулам

$$Sn_k^{(3)} = \sum_{i \in k}^K f \left( Sn_i^{(2)} \right), \quad f \left( Sn_k^{(3)} \right) = \begin{cases} 1, & \text{если } Sn_k^{(3)} > 0, \\ 0, & \text{если } Sn_k^{(3)} = 0. \end{cases}$$

Если на рис. 1 каждому распознаваемому образу соответствует один эталон, то архитектура нейронной сети преобразуется в двуслойную нейронную сеть (рис. 1, *а, б*), где количество нейронов второго слоя равно количеству распознаваемых образов  $N_{\text{обр}}$ .

### 1. Постановка задачи и методы решения

Как уже было сказано, архитектура нейронной сети на рис. 1 реализует метод ближайшего соседа, где значения весов первого слоя вычисляются аналитически по формулам (1) на основе выделенного набора эталонов. При этом напрашивается вопрос: может ли данная архитектура нейронной сети обучаться классическими алгоритмами обучения без использования выделенных эталонов и аналитических выражений?

Представим каждый нейрон на рис. 1, *а, б* первого слоя как отдельный блок нейронной сети ( $NN_{i,j}$ ) рис. 3, *а*, выполняющий задачу разделения двух образов. При этом у каждого блока  $NN_{i,j}$  имеется один бинарный выход  $y_{i,j}^{(1)} \in \{0, 1\}$ , значение которого указывает на близость объекта  $\bar{x}$  к образу  $i$  или  $j$ . Первый блок  $NN_{1,2}$  на рис. 3, *а* выполняет разделение образов 1 и 2, второй блок  $NN_{1,3}$  разделяет образы 1 и 3 и т. д., вплоть до блока  $NN_{1,N-1}$ , разделяющего образы 1 и  $N - 1$ . Затем следуют блоки  $NN_{i,j}$  для разделения пар образов  $\{2, 1\}$ ,  $\{2, 3\}$ ,  $\dots$ ,  $\{2, N - 1\}$ ,  $\{3, 1\}$  и т. д. Каждый блок  $NN_{i,j}$  обучается отдельно на разделение своей пары образов. Для обучающей выборки блока  $NN_{i,j}$  используются только объекты образов  $i, j$  (рис. 3). Для обучения блока  $NN_{i,j}$  могут использоваться любые известные классические алгоритмы обучения. Если на вход нейронной сети (рис. 3, *а*) подаётся объект  $\bar{x}$ , который принадлежит  $k$ -му образу, то на выходах первого слоя будут активны  $N - 1$  выходов, начиная от блока  $NN_{k,j}$  до блока  $NN_{k,N}$  ( $k \neq j$ ), а на выходе второго слоя (рис. 3, *б*), согласно формулам (2) и (3), будет активен выход  $k$ -го нейрона  $y_k^{(2)} = 1$ . На выходах остальных нейронов второго слоя ( $i \neq k$ ) будем иметь  $y_i^{(2)} = 0$ , поскольку для каждого  $i \neq k$  существует один блок  $NN_{i,k}$ , у которого  $y_{i,k}^{(1)} = 0$ . Таким образом, максимальное значение  $\max \left( Sn_{i \neq k}^{(2)} \right) = N - 2$ , тогда из (3) следует  $f \left( Sn_{i \neq k}^{(2)} \right) = 0$ .

В простейшем случае блок  $NN_{i,j}$  может состоять из одного нейрона. Это будет в том случае, когда разделяемые образы достаточно удалены друг от друга и легко разделяются одной гиперплоскостью. В более сложных случаях блок может состоять из нескольких нейронов и слоёв. Можно также установить фиксированное значение нейронов и слоёв для каждого блока  $NN_{i,j}$ . Нейроны блока  $NN_{i,j}$  могут быть

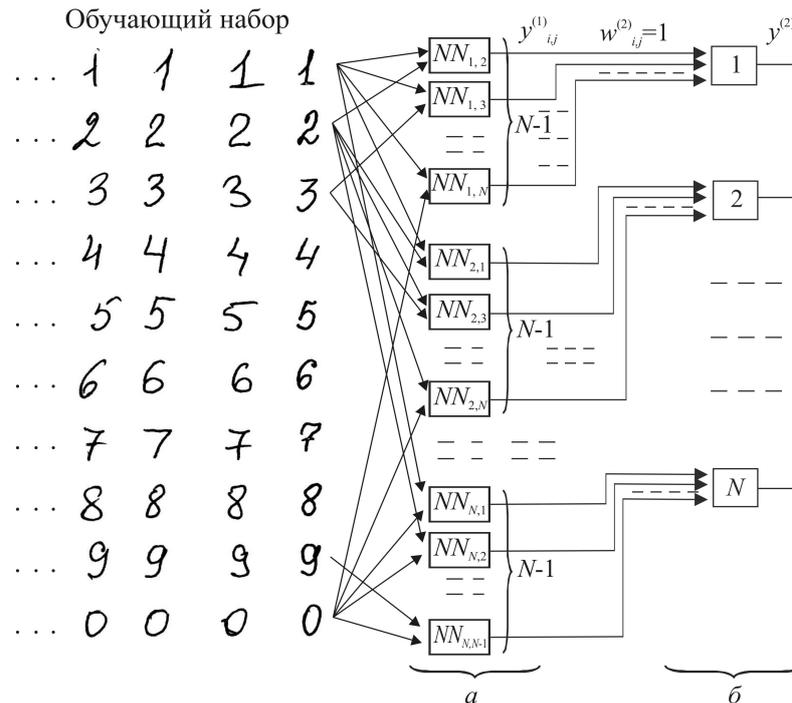
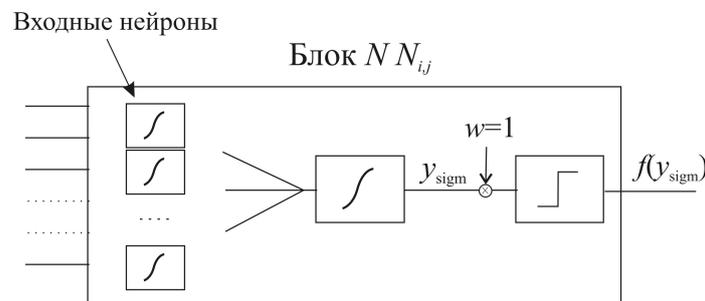


Рис. 3. Схема обучения нейронной сети

как с пороговыми, так и с непрерывными функциями активации. В последнем случае необходим дополнительный нейрон на выходе блока  $NN_{i,j}$ , преобразующий выходное непрерывное значение блока  $NN_{i,j}$  в бинарное значение  $y_{i,j}^{(1)} \in \{0, 1\}$  (рис. 4). Для этого нейрона функция активации определяется по выражению

$$f(y_{\text{sigm}}) = \begin{cases} 1, & \text{если } y_{\text{sigm}} > 0,5, \\ 0, & \text{если } y_{\text{sigm}} \leq 0,5. \end{cases}$$

В качестве алгоритмов обучения для блоков  $NN_{i,j}$  могут использоваться любые известные классические алгоритмы для нейронных сетей прямого распространения.

Рис. 4. Преобразование выхода  $NN_{i,j}$  в бинарную форму

Недостатком архитектуры нейронной сети является большое количество  $n_1$  блоков  $NN_{i,j}$ , которое для схемы на рис. 3 увеличивается по формуле  $n_1 = (N - 1) N$ . Здесь возможно уменьшить это количество в 2 раза, если исключить блоки, повторяющие разделение одноимённых пар образов, например таких, как  $\{1, 2\}$  и  $\{2, 1\}$  [7]. В этом

случае выход  $y_{i,j}^{(1)}$  блока  $NN_{i,j}$ , выполняющего разделение образов  $i$  и  $j$ , будет соединяться с  $i$ -м нейроном второго слоя ( $y_{i,j}^{(1)} = x_i^{(2)}$ ), а инвертированное значение выхода  $y_{i,j}^{(1)}$  — с  $j$ -м нейроном второго слоя ( $\bar{y}_{i,j}^{(1)} = x_j^{(2)}$ ) (рис. 5). Можно предположить, что количество блоков  $NN_{i,j}$  может быть ещё меньше, если исходить из того, что не все блоки  $NN$  являются жизненно важными и, возможно, некоторые из них могут быть удалены. Последнее утверждение требует дополнительных исследований.

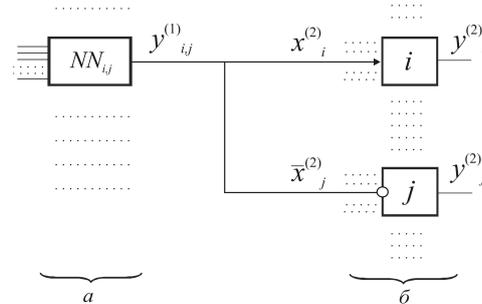


Рис. 5. Сжатие сети путём исключения блока  $NN_{ji}$

Таким образом, обучение нейронной сети (рис. 3) сводится к независимому обучению каждого отдельного блока  $NN_{i,j}$ . Обучение нейронной сети становится проще, понятнее и надёжнее, поскольку обучение множества  $N$  образов приводится к задаче попарного разделения двух образов. Исходя из этого, понадобится меньшая обучающая выборка по сравнению с классическими нейронными сетями, разделяющими много образов. Количество эпох обучения по этой же причине будет также меньше. Нет также необходимости обучать слой, приведённый на рис. 3, б. Обучение каждого блока  $NN_{i,j}$  выполняется независимо от других блоков, что позволяет легко расширять нейронную сеть, добавляя новые распознаваемые образы к общей задаче. При этом нет необходимости заново переобучать всю сеть, предыдущие весовые и пороговые значения сохраняются. Возможное количество распознаваемых образов потенциально может быть очень большим ( $N \rightarrow \infty$ ). Для каждого последующего  $(N + 1)$ -го образа к схеме нейронной сети на рис. 3 для каждого  $k$ -го образа добавляются по одному блоку  $NN_{k,N+1}$  (всего  $N$  блоков) и  $N$  блоков  $NN_{N+1,j}$  для  $(N + 1)$ -го образа. Итого  $2N$  блоков, где  $N$  — последнее значение количества образов. В случае сжатого варианта нейронной сети (рис. 5) для каждого нового образа добавляются только  $N$  новых блоков  $NN_{i,j}$ . При добавлении образа пороговое значение  $B^{(2)}$  нейрона второго слоя (3) (рис. 3, б) увеличивается на 1.

Покажем работу схемы нейронной сети на рис. 3 и 5 на примере задачи распознавания образов трёх символов «А», «В», «С». Пусть для каждого образа существует обучающая выборка с  $n$  элементами. На рис. 6 приведена сжатая схема нейронной сети (рис. 5) для данной задачи распознавания. Поскольку количество блоков на схеме равнозначно количеству нейронов первого слоя на рис. 6, б, то, исходя из выражения

$$n^{(1)} = C_N^2 = \frac{N!}{2!(N-2)!} = \frac{N(N-1)}{2} = \frac{3(3-1)}{2} = 3,$$

определяющего количество нейронов первого слоя из общего количества распознаваемых образов на основе метрического метода распознавания [7], понадобится три блока  $NN_{i,j}$ , три нейрона второго слоя и три выхода нейронной сети  $Y_i$ . Если соответствие

выходов нейронной сети к распознаваемому образу обозначено так, как на рис. 6 (в порядке «А», «В», «С» сверху вниз), то, согласно схеме, первый блок  $NN_{1,2}$  выполняет разделение образов «А», «В», блок  $NN_{1,3}$  — образов «А», «С», блок  $NN_{2,3}$  — «В», «С». На рис. 6, а каждому блоку соответствует бинарная матрица символа. При обучении на первую матрицу подаются бинарные изображения обучающих выборок образов «А», «В», на вторую — образов «А», «С» и на третью — образов «В», «С». При программной реализации данная матрица может быть в единственном числе и отдельно использоваться для каждого блока. Размерность бинарной матрицы ( $R$  — количество строк,  $C$  — столбцов) может быть различной, она определяет количество связей ( $RC$ ) для каждого входного нейрона блока  $NN_{i,j}$ . Количество входных нейронов блока  $NN_{i,j}$  может быть также различно. Чем больше размерность бинарной матрицы  $X$  и чем больше количество входных нейронов в одном блоке, тем лучше будет результат распознавания на выходе данного блока после его обучения. Каждый блок обучается отдельно при помощи известных классических алгоритмов обучения, то есть классическим способом определяется таблица весов  $W_{ij}^{(1)}$  отдельно для каждого блока  $NN_{i,j}$ . Весовые значения для нейронов второго слоя равны  $w_{ij}^{(2)} = 1$ . Поскольку используется сжатая форма нейронной сети, на входах нейронов 2 и 3 второго слоя (рис. 6, в) используются инвертированные значения выходов блоков  $y_{1,2}$ ,  $y_{1,3}$ ,  $y_{2,3}$ , которые равнозначно заменяют соответствующие входы на расширенной форме схемы нейронной сети на рис. 1. Для схемы на рис. 6 на трёх выходах нейронной сети значение «100» соответствует образу символа «А», «010» — «В» и «001» — «С».

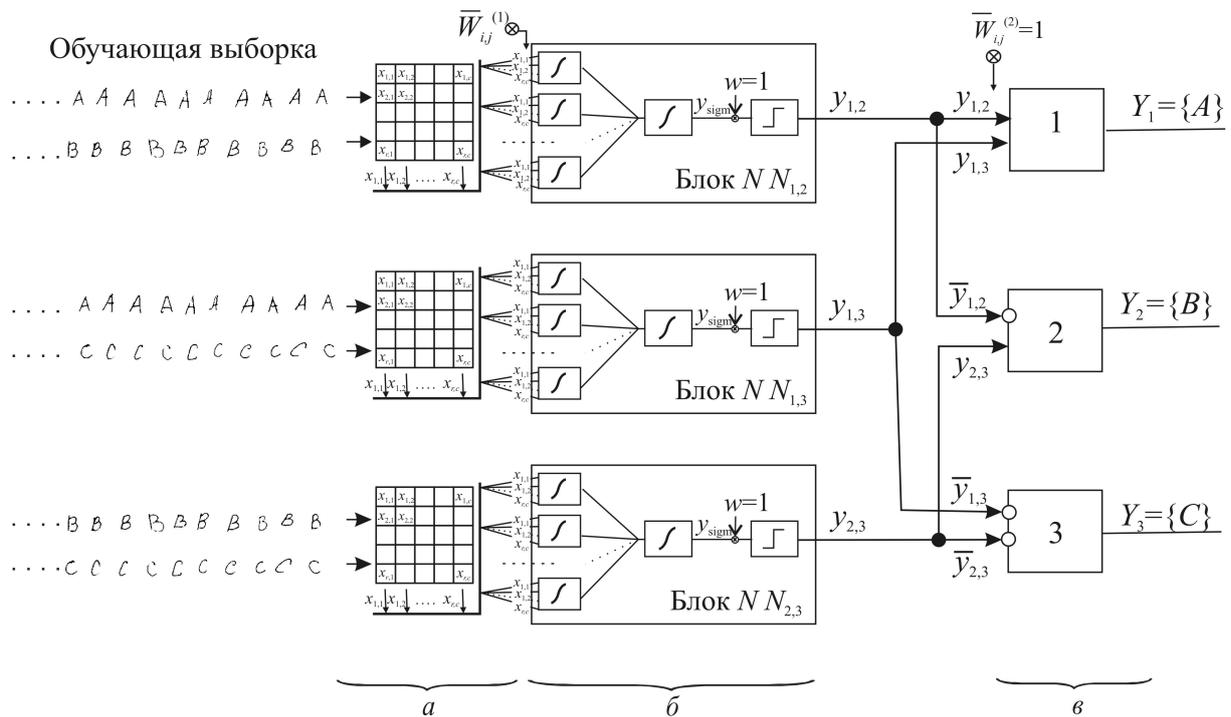


Рис. 6. Схема нейронной сети для задачи распознавания трёх образов: бинарные таблицы (а), слой блоков  $NN_{ij}$  (б), второй слой нейронной сети (в)

Предположим, что после проверки результатов обучения контрольной выборкой для каждого блока количество неверно идентифицированных символов составило для первого блока  $k_1$  символов, для второго —  $k_2$  и для третьего блока —  $k_3$  символов. В табл. 1 и 2 приведены значения состояний и выходов нейронов второго слоя (рис. 6, в)

для распознаваемого символа «А» в зависимости от значений выходов блоков  $NN_{1,2}$  и  $NN_{1,3}$ . Поскольку выход блока  $NN_{1,2}$  определяет разделение образов «А», «В» ( $y_{1,2} = 1$  для символа «А» и  $y_{1,2} = 0$  для символа «В»), а выход блока  $NN_{1,3}$  определяет разделение образов «А», «С» ( $y_{1,3} = 1$  для символа «А» и  $y_{1,3} = 0$  для символа «С»), из табл. 1 можно видеть, что для всех блоков с правильно идентифицированными символами «А» ( $y_{1,2} = 1$  и  $y_{1,3} = 1$ ) в соответствии с выражениями (2) и (3) на выходах нейронной сети формируется сигнал «100», соответствующий образу символа «А».

Таблица 1

**Значения выходов всех блоков и нейронов второго слоя для распознаваемого символа «А» для случая, когда блоки  $NN_{1,2}$  и  $NN_{1,3}$  распознают символ «А» правильно**

№ блоков	Выходы блоков	№ нейрона второго слоя	Значение нейрона второго слоя $S_n$	Выход нейрона второго слоя $Y_i$ , $N - 1 = 2$	Выбранный образ
1, 2	$y_{1,2} = 1$	1	$S_{n_1} = y_{1,2} + y_{1,3} = 1 + 1 = 2 = N - 1$	$Y_1 = 1$ (= 2)	«А»
1, 3	$y_{1,3} = 1$	2	$S_{n_2} = \overline{y_{1,2}} + y_{2,3} = 0 + (0 \text{ или } 1) = (0 \text{ или } 1) < N - 1$	$Y_2 = 0$ (< 2)	—
2, 3	$y_{2,3} = (1 \text{ или } 0)$	3	$S_{n_3} = \overline{y_{1,3}} + \overline{y_{2,3}} = 0 + (1 \text{ или } 0) = (1 \text{ или } 0) < N - 1$	$Y_3 = 0$ (< 2)	—

Таблица 2

**Значения выходов всех блоков и нейронов второго слоя для распознаваемого символа «А» для случая, когда блок  $NN_{1,2}$  распознаёт символ «А» неправильно**

№ блоков	Выходы блоков	№ нейрона второго слоя	Значение нейрона второго слоя $S_n$	Выход нейрона второго слоя $Y_i$ , $N - 1 = 2$	Выбранный образ
1, 2	$y_{1,2} = 0$	1	$S_{n_1} = y_{1,2} + y_{1,3} = 0 + 1 = 1 < N - 1$	$Y_1 = 0$ (< 2)	—
1, 3	$y_{1,3} = 1$	2	$S_{n_2} = \overline{y_{1,2}} + y_{2,3} = 1 + (0 \text{ или } 1) = (1 \text{ или } 2) \leq N - 1$	$Y_2 = 0$ (≤ 2)	«В» или —
2, 3	$y_{2,3} = (1 \text{ или } 0)$	3	$S_{n_3} = \overline{y_{1,3}} + \overline{y_{2,3}} = 0 + (0 \text{ или } 1) = (0 \text{ или } 1) < N - 1$	$Y_3 = 0$ (< 2)	—

Напротив, для случая неправильной идентификации символа «А» в блоках  $NN_{1,2}$  или  $NN_{1,3}$  на выходе итоговой нейронной сети формируется сигнал «010» или «001», соответствующий символам «В» или «С» (табл. 2). Таким образом, можно видеть, что результат работы нейронной сети (рис. 6) зависит только от правильности работы блоков  $NN_{i,j}$ , то есть, иными словами, от того, насколько хорошо обучены блоки  $NN_{i,j}$ .

Если множество ошибочно идентифицируемых символов для каждого блока обозначить как  $E(A, B)$ ,  $E(A, C)$ ,  $E(B, C)$  (рис. 7), то максимальная ошибка работы сети на контрольной выборке определяется как сумма ошибок всех блоков нейронной сети; для схемы на рис. 6  $K_{\max} = k_1 + k_2 + k_3$  в том случае, когда множества неправильно идентифицированных символов каждого образа не пересекаются друг с другом (рис. 7, а). Возможная минимальная ошибка может быть равна количеству элементов множества с максимальным количеством элементов, например  $E(A, B)$  (рис. 7, в). Для рис. 7, б ошибка  $K$  определяется количеством элементов множества  $E$ , которое формируется по выражению

$$|E| = |E(A, B)| + |E(A, C)| - |E(A, B) \cap E(A, C)| + |E(B, C)| - |E(A, B) \cap E(B, C)|.$$

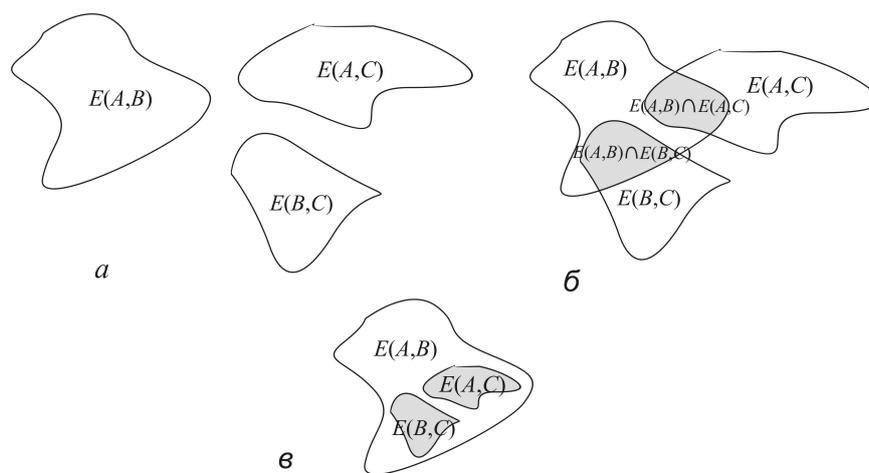


Рис. 7. Непересекающиеся множества ошибок для каждого блока (а); пересекающиеся множества (б) и (в)

Нужно также отметить, что, в отличие от схемы на рис. 1, в которой входные данные представляются в виде бинарной матрицы, для схем на рис. 3 и 6 значения входных матриц могут быть отличны от 0 и 1, поскольку попарное разделение двух образов выполняется классической схемой нейронной сети. Это позволяет в качестве входных данных в схемах на рис. 3 и 6 использовать чёрно-белые растровые изображения, у которых каждый пиксель может принимать значение в диапазоне  $0 \div 256$ ; например, на рис. 8 показано изображение символа «8» со значениями пикселей в серых оттенках.

Матрица на рис. 8 может быть нормирована, например, в диапазон  $(0, 1)$ . Могут быть также использованы отдельные палитры цветных изображений, представленные в той или иной цветовой схеме, например RGB, CMYK и т. д. Это может позволить в качестве входных данных нейронных сетей на рис. 3 и 6 использовать цветные изображения.

Само представление входных данных может быть отлично от непосредственного использования пикселей изображения. В качестве входных данных могут быть использованы предварительно определённые, выделенные или вычисленные данные изображения, которые более информативны для решаемой задачи. Например, в работе [10] приведены подходы представления входных данных для различных задач, в частности возможные способы представления данных для задачи распознавания лиц, а также для задачи распознавания символов с использованием инвариантных способов представления символов [11], позволяющие узнавать изображения символов с учётом возможных сдвигов, поворотов и масштабирования данного изображения.



свойственно биологическому мозгу. Это отличает данную архитектуру, например, от архитектуры сверточных сетей [4–6], где объекты последовательно (последовательно) анализируются от простейших деталей изображения до сложных. Кроме того, в предлагаемой архитектуре количество распознаваемых или добавляемых образов практически не ограничено, что также очень похоже на возможности биологического мозга, который может легко запоминать и узнавать огромное количество объектов и образов.

#### ЛИТЕРАТУРА

1. Барский А. Б. Нейронные сети: распознавание, управление, принятие решений. М.: Финансы и статистика, 2004.
2. Головки В. Л. Нейронные сети: обучение, организация и применение / под ред. А. И. Галушкина. Кн. 4. М.: ИПРЖР, 2001.
3. Уоссерман Ф. Нейрокомпьютерная техника. Теория и практика. М.: Мир, 1992.
4. LeCun Y., Bengio Y., and Hinton G. Deep learning // Nature. 2015. No. 521. P. 436–444.
5. Schmidhuber J. Deep learning in neural networks: An overview // Neural Networks. 2015. No. 61. P. 85–117.
6. <http://www.neuralnetworksanddeeplearning.com/index.html> — Neural Networks and Deep Learning.
7. Geidarov P. Sh. Neural networks on the basis of the sample method // Automatic Control and Computer Sci. 2009. V. 43. No. 4. P. 203–210.
8. Geidarov P. Sh. Multitasking application of neural networks implementing metric methods of recognition // Autom. Remote Control. 2013. V. 74. No. 9. P. 1474–1485.
9. Биргер И. А. Техническая диагностика. М.: Машиностроение, 1978.
10. Гейдаров П. Ш. Алгоритм реализации метода ближайшего соседа в многослойном персептроне // Труды СПИИРАН. 2017. Вып. 51. С. 123–151.
11. Васин Д. Ю., Аратский А. В. Распознавание символов на основе инвариантных моментов графических изображений // 25 Междунар. конф. GraphiCon 2015. М., 2015. С. 259–264.

#### REFERENCES

1. Barskiy A. B. Neyronnye seti: raspoznavanie, upravlenie, prinyatie resheniy [Neural Networks: Recognition, Management, Decision Making]. Moscow, Finansy i Statistika Publ., 2004. (in Russian)
2. Golovko V. L. Nejronnye seti: obuchenie, organizacija i primenenie [Neural Networks: Training, Organization and Application]. Book 4, Moscow, IPRJR Publ., 2001. (in Russian)
3. Wasserman F. Neyrokomp'yuternaya tekhnika. Teoriya i praktika [Neurocomputing. Theory and Practice]. Moscow, Mir Publ., 1992. (in Russian)
4. LeCun Y., Bengio Y., and Hinton G. Deep learning. Nature, 2015, no. 521, pp. 436–444.
5. Schmidhuber J. Deep learning in neural networks: An overview. Neural Networks, 2015, no. 61, pp. 85–117.
6. <http://www.neuralnetworksanddeeplearning.com/index.html> — Neural Networks and Deep Learning.
7. Geidarov P. Sh. Neural networks on the basis of the sample method. Automatic Control and Computer Sci., 2009, vol. 43, no. 4, pp. 203–210.
8. Geidarov P. Sh. Multitasking application of neural networks implementing metric methods of recognition. Autom. Remote Control, 2013, vol. 74, no. 9, pp. 1474–1485.
9. Birger I. A. Tekhnicheskaya diagnostika [Technical Diagnostics]. Moscow, Mashinostroenie Publ., 1978. (in Russian)

10. *Geidarov P. Sh.* Algoritm realizacii metoda blizhajshego soseda v mnogoslojnom perseptrone [Algorithm for realizing the method of the nearest neighbor in a multilayer perceptron]. Trudy SPIIRAN, 2017, no. 51, pp. 123–151. (in Russian)
11. *Vasin D. Yu. and Aratskiy A. V.* Raspoznavanie simvolov na osnove invariantnyh momentov graficheskikh izobrazhenij [Character recognition based on invariant moments of graphic images]. 25th Intern. GraphiCon Conf., Moscow, 2015, pp. 259–264. (in Russian)