

## **НЕЙРОСЕТИ НА МНОГОЗНАЧНЫХ НЕЙРОЭЛЕМЕНТАХ: ОБУЧЕНИЕ ОБРАБОТКА И РАСПОЗНАВАНИЕ ИЗОБРАЖЕНИЙ**

В данной работе рассматривается развитие концепции многозначных нейроэлементов. Прежде всего предлагается существенно усиленный алгоритм обучения, позволяющий ускорить процесс обучения в 20-30 раз и добиться сходимости обучения даже для таких функций, для которых ранее успешное обучение считалось невозможным. Рассматриваются сети из многозначных нейроэлементов, ориентированные прежде всего на решение задач обработки (клеточные сети) и распознавания (сети со случайными связями) изображений. Показано, как на указанных сетях решаются задачи выделения контуров, анализа текстур, построения ассоциативной памяти для полутоновых изображений.

### **1. ВВЕДЕНИЕ**

Многозначные нейроэлементы, основанные на концепции многозначных пороговых функций  $k$ -значной логики [1] были впервые предложены в [2], а затем различные аспекты их теории были развиты в [3] и [4]. Основной особенностью многозначных нейроэлементов является представление обрабатываемой информации и весов нейроэлементов с помощью комплексных чисел. На первый взгляд такое представление может показаться неестественным и слишком громоздким в чисто вычислительном аспекте, но как было показано в [1-4] оно позволяет определить выходную (активирующую) функцию нейрона таким образом, что алгоритм обучения для многозначного нейрона сходится в сотни раз быстрее, чем алгоритмы обучения для других типов нейронов, а, кроме того, функциональные возможности многозначных нейронов значительно выше, чем у других типов нейронов.

Применение нейросетей для решения задач обработки изображений вошло в широкую практику с конца 80-х годов после публикации работы Чуа-Янга [5], в которой была предложена концепция клеточной сети, весьма удобная для реализации, например различных алгоритмов линейной фильтрации. В [2] нами была предложена клеточная сеть на многозначных нейронах и ее использование в качестве ассоциативной памяти для хранения полутоновых изображений. В [3] и [4] мы рассматривали применение такой же клеточной сети (а также клеточной сети из универсальных бинарных нейронов [6], математическая основа которых та же, что и у многозначных, но предназначены они для реализации не многозначных пороговых, а произвольных булевых функций) для решения некоторых задач обработки изображений (фильтрация импульсного шума, выделение контуров). Главная цель настоящей работы - дальнейшее развитие теории многозначных нейронов и расширение областей их применения. Мы рассмотрим алгоритм обучения, обоснованный в [1] и [3], но с существенным усилением, позволяющим более чем в 30 раз ускорить процесс сходимости алгоритма обучения, а также увеличить число входов нейрона и уровней обрабатываемых сигналов с десятков и сотен до тысяч. Мы рассмотрим также новый тип сети из многозначных элементов - сеть со случайными связями, в которой каждый нейроэлемент соединен с фиксированным числом других, но не из ближайшей окрестности, как в случае клеточных сетей, а с элементами, определяемыми некоторой случайной функцией. На базе этой сети будет предложена ассоциативная память, весьма эффективная для хранения и распознавания полутоновых изображений. Будет также рассмотрено использование клеточной сети из многозначных и универсальных бинарных нейронов для решения задач выделения контуров на полутоновых и бинарных изображениях, а также для анализа текстур.

## 2. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ

Сначала напомним некоторые основные понятия теории многозначных и универсальных нейроэлементов. Эти нейроны реализуют полностью определенные пороговые  $k$ -значные функции, либо частично определенные  $k$ -значные функции (многозначные нейроны) и произвольные булевы функции (универсальные бинарные нейроны) от  $n$  переменных  $f(x_1, \dots, x_n)$  путем их представления при помощи  $n+1$  комплексных параметров (весов)  $w_0, w_1, \dots, w_n$

$$f(x_1, \dots, x_n) = P(w_0 + w_1 x_1 + w_n x_n) \quad (1)$$

где  $x_1, \dots, x_n$  - переменные, от которых зависит реализуемая функция (входы нейрона) (отметим, что значения функции и ее переменных в многозначном ( $k$ -значном) случае кодируются комплексными числами, являющимися корнями  $k$ -ой степени из единицы:  $\varepsilon^j = \exp(i 2\pi j / k)$ ,  $j \in [0, k - 1]$ ,  $i$  - мнимая единица; иначе говоря, значения  $k$ -значной логики представляются корнями  $k$ -ой степени из единицы:  $\varepsilon^j \rightarrow \varepsilon$ ), а  $P$  - выходная (активирующая) функция нейрона, определяемая для многозначного нейроэлемента:

$$P(z) = \exp(i * 2\pi * j / k),$$

$$\text{if } 2\pi * (j + 1) / k > \arg(z) \geq 2\pi * j / k \quad (2a)$$

где  $j=0,1,\dots,k-1$  - значения в  $k$ -значной логике,  $i$  - мнимая единица  $z = w_0 + w_1 x_1 + w_n x_n$  - взвешенная сумма,  $\arg(z)$  - аргумент комплексного числа  $z$ ; и для универсального бинарного нейроэлемента:

$$P(z) = (-1)^j$$

$$\text{if } (2\pi (j + 1) / m) > \arg(z) \geq (2\pi * j / m) \quad (2b)$$

где  $m$  - некоторое натуральное число,  $0 \leq j < m$ ,  $z = w_0 + w_1 x_1 + w_n x_n$ ,  $\arg(z)$  - аргумент комплексного числа  $z$ . Переменные (входы нейрона) и реализуемая функция в данном случае булевы и принимают значения в множестве  $\{1, -1\}$ . Идея подобного представления многозначных функций была впервые предложена в [1], а булевых - в [6]. Определение выходных функций нейроэлементов (2a) и (2b) иллюстрируется на рис. 1a и 1b соответственно:

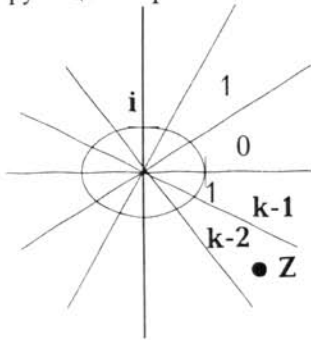


Рис.1a

$$P(z) = \varepsilon^{k-2}$$

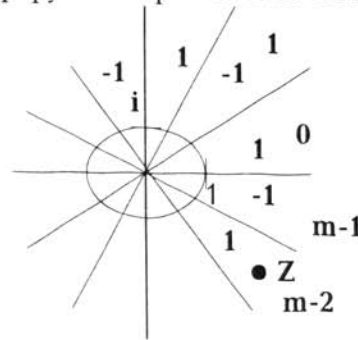


Рис.1b

$$P(z) =$$

## 3. УСОВЕРШЕНСТВОВАННЫЙ АЛГОРИТМ ОБУЧЕНИЯ

Из описанной математической модели очевидно, что обучение для обоих рассматриваемых типов нейронов состоит в "направлении" взвешенной суммы входных сигналов в нужный сектор на комплексной плоскости (из  $k$  возможных для многозначного и из  $m$  возможных для универсального бинарного нейрона) (Рис.1). Алгоритм обучения для многозначного нейроэлемента был предложен в [1], и затем ряд его модификаций был предложен в [2-4]. Многочисленные

эксперименты показали, что скорость сходимости этого алгоритма существенно выше скорости сходимости алгоритмов обучения для нейронов Хопфилда, Чуа и других подобных "классических" нейроэлементов.

Процедура "направления" взвешенной суммы в нужный сектор комплексной плоскости (иначе говоря, процедура коррекции вектора структуры) определяется следующим правилом [1]:

$$\mathbf{W}_{m+1} = \mathbf{W}_m + \omega \varepsilon^q \bar{\mathbf{X}}, \quad (3)$$

где  $\mathbf{W}_m$  and  $\mathbf{W}_{m+1}$  - текущий и следующий весовые векторы (векторы структуры),  $\omega$  - коэффициент коррекции,  $\bar{\mathbf{X}}$  - вектор входных сигналов нейрона с комплексно-сопряженными компонентами,  $\varepsilon^q$  - значение функции на выходе (выходной сигнал нейрона), таким образом,  $q$  - номер нужного сектора (иначе говоря,  $q$  - целочисленное значение  $k$ -значной функции, реализуемой нейроном).  $\mathbf{W}$  выбирается так, чтобы значение взвешенной суммы  $Z = W_0 + W_1 X_1 + W_n X_n$  было как можно ближе к нужному сектору (либо сразу же попадало в него) (к сектору с номером  $q$ ) после шага обучения, определяемого правилом (3). Как правило, достаточно рассмотреть всего два случая для выбора  $\mathbf{w}$  -  $|\arg(\varepsilon^q) - \arg(z)| \leq \pi / 2$  или  $|\arg(\varepsilon^q) - \arg(z)| > \pi / 2$ . Если выполняется первое из этих условий, как правило достаточно одной-двух итераций для попадания в нужный сектор по правилу (3). Для второго случая число итераций, как правило, несколько больше, но более чем на порядок. Под "итерацией" следует понимать последовательную проверку (1) для всех элементов обучающего множества, и в том случае, если (1) не выполняется для некоторого элемента из этого множества, для коррекции весов применяется обучающая процедура (3). Этот процесс продолжается до тех пор, пока не будет получен вектор структуры  $\mathbf{W} = (W_0, W_1, \dots, W_n)$ , удовлетворяющий (1) для всех элементов обучающего множества. Как показывают экспериментальные данные, процесс обучения, определяемый правилом (3), сходится для большинства частично определенных функций  $k$ -значной логики. В [1] доказано, что процесс обучения сходится также для полностью определенных многозначных пороговых функций. Даже в случае несходящегося процесса для некоторой функции (в частности, для частично определенной функции) практически всегда можно так изменить значение  $k$ , чтобы свести нереализуемую на нейроне функцию  $k$ -значной логики к реализуемой функции  $\tilde{k}$ -значной логики ( $k < \tilde{k}$  или  $\tilde{k} > k$ ), для которой процесс обучения сходится.

Непосредственное применение арифметики с плавающей запятой к вычислениям с комплексными числами в (3) позволяет реализовать алгоритм обучения, однако с ростом  $k$  и  $n$  (числа переменных) вычисления все более замедляются и, например для обучения сети из 4096 нейронов (каждый из них имеет 30 входов,  $k=256$ , обучающее множество для каждого нейрона состоит из 30 элементов) требуется 70 часов на компьютере класса 486-DX2-66 Mhz. Модификация алгоритма обучения, которая сейчас будет рассмотрена, позволяет с одной стороны ускорить процесс обучения примерно в 30 раз, а с другой стороны обучать нейроны с числом входов, равным  $n = 1000-2000$  реализации функций  $k = 2000-4000$  - значной логики.

Прежде всего откажемся от вычислений с плавающей запятой. Действительную и мнимую часть весов будем представлять в виде 32-битовых целых чисел, а взвешенную сумму - 64-битовым целым числом. Для ускорения вычисления выходного значения нейроэлемента, т.е., для ускорения вычислений в (2а) необходимо наложить несущественное ограничение на возможные значения  $k$ , а именно,  $k$  должно быть кратно 8. Несущественность этого ограничения очевидна, т.к. при необходимости значение  $k$  всегда можно увеличить до необходимого. Зато благодаря этому ограничению мы можем вычислить  $j$  (целочисленное значение функции на выходе нейрона, соответствующее комплексному  $\varepsilon$ ), прибегая к существенно меньшим вычислительным затратам.

Пусть  $z = (\text{Re}(z), \text{Im}(z))$  - комплексное число, представляющее взвешенную сумму и

$$\begin{cases} \text{abs}(\text{Re}(z) / \text{abs}(\text{Im}(z))), & \text{если } \text{abs}(\text{Re}(z)) > \text{abs}(\text{Im}(z)) \\ \text{abs}(\text{Im}(z) / \text{abs}(\text{Re}(z))), & \text{если } \text{abs}(\text{Im}(z)) < \text{abs}(\text{Re}(z)) \end{cases} \quad (4)$$

Вместо вычисления  $\arg(z)$  путем многочисленных сравнений аргументов взвешенной суммы и корней соответствующей степени из единицы мы будем вычислять значение  $j$  функции на выходе нейрона непосредственно, пользуясь заранее созданной таблицей:

$$(l) = \text{tg}(2\pi l / k), \text{ где } l = 0, 1, \dots, (k / 8) - 1 \quad (5)$$

Из (4), (5) и Рис. 2 очевидно, что очень легко может быть найдено такое значение  $l$ , что  $(\mathbf{I}) \leq \phi \leq \mathbf{T}(\mathbf{I})$ . Принимая во внимание знак и значение отношения  $\text{Im}(z)/\text{Re}(z)$ , легко найти номер  $s$  "подсектора", в который попадает  $z$ . Теперь очевидно, что

$$j = s(k/8) + l, \quad s = 0, 1, \dots, (k/8) - 1, \quad (6)$$

где  $j$  - значение на выходе нейрона.

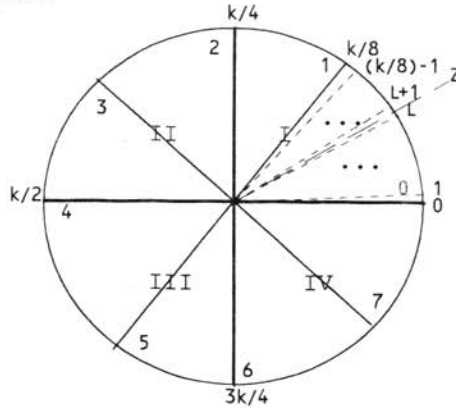


Рис.2

Теперь вернемся к примеру с сетью, содержащей 4096 30-входных нейронов, для которой решается задача обучения на обучающем множестве, содержащем 30 элементов (мы еще раз вернемся ниже к применению этой сети). Использование того же правила обучения (3), но с учетом (4)-(6) позволяют решить задачу обучения за 2 часа 25 минут на том же компьютере класса 486 DX2-66 Mhz (без использования модификации (4)-(6) та же задача решается за 70 часов).

Следует сделать ряд замечаний по поводу обучения универсального бинарного нейрона. Очевидны два подхода к решению этой задачи. Первый основывается на непосредственном использовании процедуры (3).  $\varepsilon^q$  в этом случае соответствует правой границе того сектора на комплексной плоскости, в который должна попасть взвешенная сумма  $Z = W_0 + W_1 X_1 + W_n X_n$  (Рис. 1a). Если  $Z$  попадает "не в тот" сектор, веса должны быть скорректированы в соответствии с (3), чтобы направить  $Z$  в соседний сектор. Исходя из этих соображений, значения коэффициента коррекции  $\Omega$  в (3) определяются следующим образом (i - мнимая единица):

$$\begin{cases} 1, & \text{if } Z \text{ должно "переместиться" в правый сектор} \\ i, & \text{if } Z \text{ должно "переместиться" в левый сектор} \end{cases} \quad (7)$$

Экспериментально показано, что подобный процесс обучения довольно быстро сходится прежде всего для достаточно малых  $m$  (см. (2b)) (например, при  $m=4$  для функций, допускающих представление (1) достаточно как правило не более 100 итераций для сходимости алгоритма обучения). Другой подход к обучению универсального бинарного нейрона сводится к преобразованию булевой функции в  $m$ -значную (каждая "1" преобразуется в некоторое четное значение, каждая "-1" - в некоторое нечетное) и затем к применению алгоритма обучения (3)-(6) к полученной многозначной функции. Однако, подробное обсуждение этой проблемы выходит за рамки данной работы. Один из подходов к ее решению предложен в [4].

#### 4. СЕТИ ИЗ МНОГОЗНАЧНЫХ И УНИВЕРСАЛЬНЫХ ЭЛЕМЕНТОВ И ИХ ПРИМЕНЕНИЕ ДЛЯ ОБРАБОТКИ И РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЙ

Обе разновидности рассматриваемого нейроэлемента очень удобны для создания сетей, предназначенных в первую очередь для обработки и распознавания изображений. Мы предлагаем использовать два типа сетей. Первая из них клеточная, в которой каждый нейрон соединен только с нейронами из своей ближайшей прямоугольной (квадратной) окрестности. Хотим, однако, обратить внимание, что это не клеточная сеть Чуа [5], т.к. базисные нейроны нашей сети и сети Чуа совершенно различны. Второй тип сети, который мы хотим предложить, это сеть со случайными связями (каждый нейрон в такой сети соединен, как и в клеточной сети, с ограниченным числом других нейронов, но их координаты определяются некоторой случайной



функцией). На Рис. 3а показан пример клеточной сети с локальной окрестностью 3\*3, а на Рис. 3б

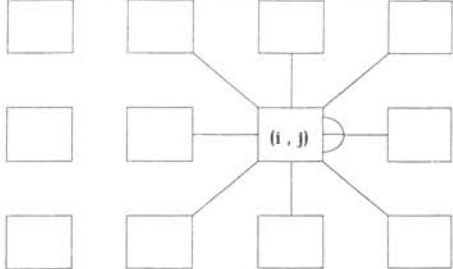


Рис.3а

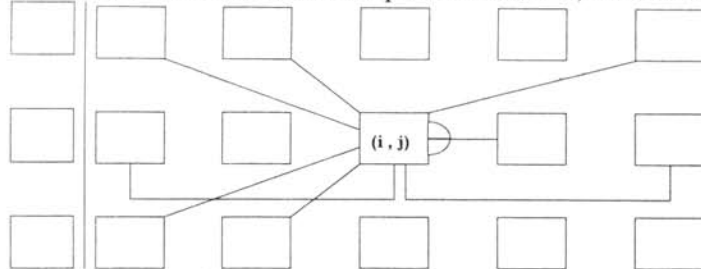


Рис.3б

показан пример сети со случайными связями, в которой каждый (ij-тый) нейрон соединен с восемью другими и имеет обратную связь.

### ВЫДЕЛЕНИЕ КОНТУРОВ

Первым, весьма эффективным и эффективным применением клеточной сети является решение на ней задачи выделения контуров на полутоновых изображениях. В качестве базового элемента сети будем использовать универсальный бинарный нейроэлемент. Алгоритм выделения контуров состоит из следующих шагов: 1.Полутоновое изображение с 256-ю градациями яркости разбивается на 8 битовых плоскостей; 2.На каждой битовой плоскости, представляющей собой бинарное изображение, производится выделение контуров; 3.Осуществляется "сшивание" бинарных контурных изображений в полутоновое.

Выделение контуров на бинарном изображении исчерпывающим образом описывается следующей булевой функцией :

$$Y \begin{bmatrix} X_1 & X_2 & X_3 \\ X_4 & X_5 & X_6 \\ X_7 & X_8 & X_9 \end{bmatrix} = \begin{cases} -1, & \text{если } x_5 = -1 \text{ или } x_1 = \dots = x_5 = \dots = x_9 = 1 \\ 1, & \text{если } x_5 = 1 \text{ и хотя бы один из } x_1, \dots, x_4, x_6, \dots, x_9 \neq 1 \end{cases} \quad (8)$$

другими словами, контур регистрируется в данной точке, если он проходит через центральную точку окна 3\*3, окружающего данную точку. Пусть m=4 в (2б). Применение алгоритма обучения (7) к функции (8) дает сходимость после 51 итерации (на которые затрачивается всего несколько секунд на компьютере класса 486 DX2-66 Mhz. Получив в результате обучения необходимый набор весов, мы можем решать задачу выделения контуров на нейросети. Отметим, что при аппаратной реализации сети можно сделать ее восьмиуровневой, чтобы обрабатывать каждую битовую плоскость параллельно в режиме реального времени. На рисунках 4а и 5а показаны оригинальные изображения, на которых проводилось выделение контуров описываемым методом. На рисунках 4б и 5б показан результат обработки, а на рисунках 4с и 5с - результат выделения контуров на тех же изображениях при помощи оператора Лапласа. Преимущества описанного здесь подхода очевидны хотя бы из того, что он позволяет выделить как малейшие пе-



Рис.4а



Рис.4б



Рис.4с

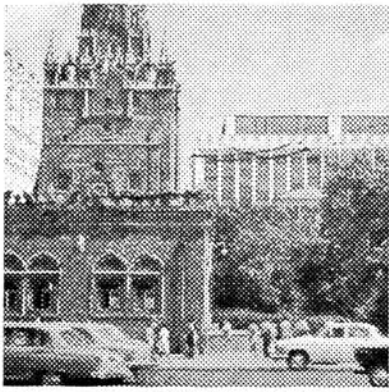


Рис.5а

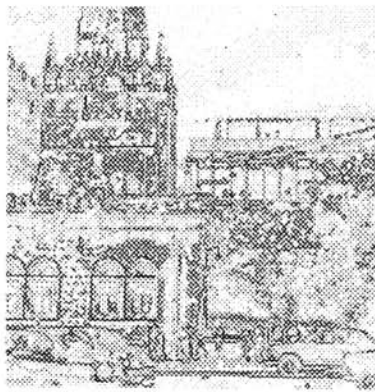


Рис.5б

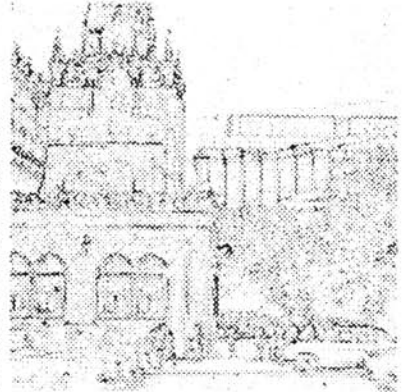


Рис.5с

репады яркостей, так и даже контуры "смазанных", нерезких деталей.

### СЕГМЕНТАЦИЯ И АНАЛИЗ ТЕКСТУР

На такой же клеточной сети, но построенной из многозначных нейронов, можно эффективно решать задачу анализа текстур. Для этого, очевидно, следует в процессе обучения добиться того, чтобы многозначный нейроэлемент различал необходимые текстуры. Естественно, результат будет тем лучше, чем большие по площади участки эталонных текстур будут присутствовать в обучающей выборке. При этом естественно следует учесть, что время, затрачиваемое на обучение, будет увеличиваться. Так что, следует искать "золотую середину" между мощностью обучающего множества и временем обучения. Проблема существенно упрощается при использовании высокопроизводительной вычислительной техники типа рабочих станций. На рис. 6 представлены два текстурных изображения размера  $64 \times 64$ , составивших обучающее множество. Первое из этих изображений представляет собой поле из случайных чисел с равномерным, а второе - с гауссовским распределением. Обучение многозначного нейроэлемента с 91-м входом (нейрон соединен со всеми нейронами из своей локальной окрестности  $9 \times 9$ ) по алгоритму (3)-(6) на компьютере класса 486-DX2-66Mhz потребовало около 60 часов. Для проверки обученной сети было предложено проанализировать изображение на рис.7, где на равномерное поле "наложен" участок гауссовского. Результат анализа представлен изображением на рис.8

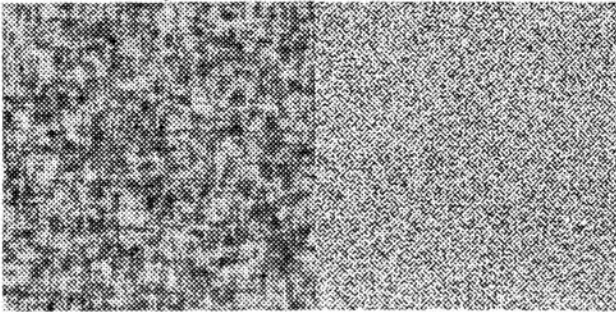


Рис.6а

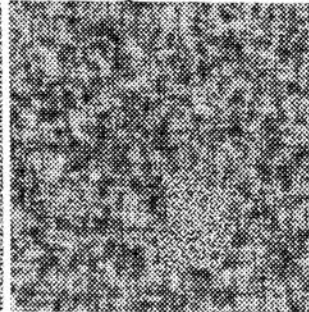


Рис.6б

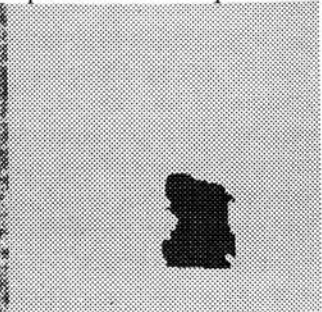


Рис.7

Рис.8

Как видим, разделение текстур произошло абсолютно правильно. Очевидным достоинством данного метода является также высокая толерантность обученного нейроэлемента (или сети - в данном случае, как и в случае выделения контуров все нейроны сети реализуют одну и ту же функцию) к поворотам эталонных образов.

### РАСПОЗНАВАНИЕ И ХРАНЕНИЕ ИЗОБРАЖЕНИЙ В АССОЦИАТИВНОЙ ПАМЯТИ

Ассоциативная память, т.е., память, адресуемая по содержанию, является эффективным средством не только для хранения данных, но и для их распознавания, точнее - узнавания в условиях зашумленности, частичного искажения и т.д. Создание подобных систем памяти, базирующихся на нейроподобных сетях, для полутоновых изображений впервые было предложено Т.Кохоненом [7]. В [2] нами была предложена модель ассоциативной памяти для полутоновых изображений, базирующейся на клеточной сети из многозначных нейронов. Недостатком такого подхода является невозможность узнавания изображения в случае полного разрушения какого-то его фрагмента, поскольку ограниченность связей в клеточной сети не позволяет восстановить полностью утраченный фрагмент. Этот недостаток преодолевается, если в качестве базовой сети для ассоциативной памяти избрать сеть со случайными связями, описанную выше.

Для проверки работы подобной ассоциативной памяти была разработана программная модель сети, содержащей 4096 многозначных (в данном случае - 256-значных) нейроэлементов с 30

входами каждый. Множество нейронов, соединенных с каждым данным определялось с помощью датчика случайных чисел (вертикальные и горизонтальные координаты отдельно). Обучающее множество содержало 40 полутоновых изображений с 256-ю градациями яркости различных крупноплановых портретов (четыре из них представлены на рис.9) размера 64\*64. Не следует



Рис.9

думать, что на размеры наложено какое-то ограничение, поскольку в данном случае выбор размера определялся исключительно желанием ускорить обучение в условиях моделирования сети на обычном компьютере класса 486 DX2-66. Обучение всех нейронов сети, построенное по алгоритму (3)-(6), занимает 3 часа 20 минут при наличии у каждого нейрона обратной связи (один вход соединен с собственным выходом) и 5 часов 15 минут без обратной связи. Отметим, что наличие или отсутствие обратной связи, как показывают эксперименты, не оказывает в данном случае решающего влияния на свойства ассоциативной памяти. Отличные же возможности памяти демонстрируют эксперименты, иллюстрируемые на рис. 10 и 11. На рис.10а представлено одно из изображений обучающей выборки, в котором 75% информации заменено равномерным шумом. На рис.10b - результат восстановления (узнавания) изображения после 20-ти итераций работы сети (под итерацией понимается срабатывание всех нейронов сети), на рис.10с - результат

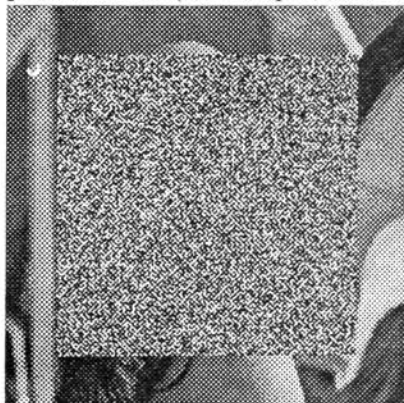


Рис.10а



Рис.10b



Рис.10с

восстановления изображения после 40 итераций работы сети. Очевидно, что уже качества изображения после 20-ти итераций достаточно для его распознавания, но после 40 итераций полученное изображение вообще практически не отличается от оригинала (среднеквадратическое отклонение - 0.87). На рис.11а показано то же изображение из обучающей выборки, "испорченное" импульсным шумом с вероятностью искажения отсчетов 0.3. На рис.11b - результат восстановления изображения после 35-ти, а на рис. 11с - после 70-ти итераций работы



Рис.11а



Рис.11b



Рис.11с

сети. Вновь промежуточного результата вполне достаточно для узнавания изображения, а окончательный результат восстановления вновь практически не отличается от оригинала. Значительно более высокая эффективность данного подхода по сравнению с изложенным в [8], где



на изображениях того же размера допускалось искажение не более 25 % информации (иначе оно не восстанавливалось) не вызывает сомнений.

## 5. ЗАКЛЮЧЕНИЕ

Основной вывод, который можно сделать по результатам данной работы - это весьма высокая эффективность многозначных нейроэлементов (и их разновидности - универсальных бинарных нейроэлементов) по сравнению с другими типами нейроэлементов прежде всего благодаря алгоритму обучения, свободному как от решения сложных систем линейных неравенств, так и систем дифференциальных уравнений (либо то, либо другое необходимо для любых других типов нейроэлементов). Наличие эффективного алгоритма обучения позволяет решать с помощью различных типов сетей, построенных из рассматриваемых нейроэлементов, широкий круг задач, и в частности, как было показано, весьма эффективно решать задачи обработки и распознавания изображений: выделения контуров, анализ текстур, хранение и распознавание изображений в ассоциативной памяти, получая при этом результаты, как превосходящие те, что получаются при применении других методов решения тех же задач, так и те, что получаются при применении других типов нейросетей и иных базовых нейроэлементов.

## ЛИТЕРАТУРА

1. Айзенберг Н.Н., Иваськив Ю.Л. Многозначная пороговая логика. Киев: Наукова Думка, 1977.
2. Aizenberg N.N., Aizenberg I.N. "CNN Based on Multi-Valued Neuron as a Model of Associative Memory for Gray-Scale Images". Proc. of the 2-d International IEEE Workshop on Cellular Neural Networks and their Applications, Munich, Germany, October 14-16, 1992, IEEE 92TH0498-6, ISBN 0-7803-875-1, pp.36-41.
3. Aizenberg N.N., Aizenberg I.N. "Fast Converged Learning Algorithms for Multi-Level and Universal Binary Neurons and Solving of the some Image Processing Problems", Lecture Notes in Computer Science, Ed.-J.Mira, J.Cabestany, A.Prieto, v.686, Shpringer-Verlag, Berlin-Heidelberg (1993) pp. 230-236.
4. Aizenberg N.N., Aizenberg I.N. "CNN-like Networks Based on Multi-Valued and Universal Binary Neurons: Learning and Application to Image Processing", Proc. of the Third IEEE International Workshop on Cellular Neural Networks and their Applications, Roma, Italy, December 18-21, 1994, IEEE 94TH0693-2, pp.153-158.
5. L.O. Chua and L. Yang, "Cellular neural networks: Theory", IEEE Trans.Circuits Syst.vol. 35, pp. 1257-1290, Oct. 1988
6. Айзенберг И.Н. Универсальный логический элемент над полем комплексных чисел//Кибернетика, № 3, 1991, С. 116-121.
7. Т.Kohonen "Content-Addressable Memories" Springer -Verlag, Berlin - Heidelberg - New-York, 1980.
8. U.Ramaher, W.Raab, J.Anlauf, U.Hachmann, J.Beichter, N.Bruls, M.Weseling, E.Sichender, R.Manner, J.Glass, A.Wurz "Multiprocessor and Memory Architecture of the Neurocomputer Synapse-1" Proceedings of the 3-d International Conference on Microelectronics for Neural Networks. April 6-8, 1993, Edinburgh, UK, pp. 227-231.



# Neural networks based on multi-valued neural elements: learning, image processing and recognition

*N.N. Aizenberg, I.N. Aizenberg, G.A. Krivosheev*

## *Abstract*

This work considers the development of the concept of multi-valued neural elements. First of all, a significantly enhanced learning algorithm is proposed, which allows to accelerate the learning process by 20-30 times and to achieve training convergence even for the functions for which the successful learning used to be considered impossible. We consider the networks of multi-valued neural elements, focused primarily on solving the issues of processing (cell networks) and recognition (networks with random connections) of images. We show the ways to solve the tasks of detecting edges, analyzing textures, and developing associative memory of half-tone images using the above networks.

*Citation:* Aizenberg NN, Aizenberg IN, Krivosheev GA. Neural networks based on multi-valued neural elements: learning, image processing and recognition. *Computer Optics* 1995; 14-15(1): 179-186.

## *References*

- [1] Aizenberg NN, Ivaskiv YL. Multiple-valued threshold logic, Kiev: Naukova Dumka; 1977
- [2] Aizenberg NN, Aizenberg IN. CNN Based on Multi-Valued Neuron as a Model of Associative Memory for Gray-Scale Images. Proc. of the 2-d International IEEE Workshop on Cellular Neural Networks and their Applications; Germany; Munich: 1992; IEEE 92TH0498-6; ISBN 0-7803-875-1; 36-41.
- [3] Aizenberg NN, Aizenberg IN. Fast Converged Learning Algorithms for Multi-Level and Universal Binary Neurons and Solving of the some Image Processing Problems, Lecture Notes in Computer Science; Ed. Mira J, Cabestany J, Prieto A. Shpringer-Verlag, Berlin-Heidelberg: 1993, 686: 230-236.
- [4] Aizenberg NN, Aizenberg IN. CNN-like Networks Based on Multi-Valued and Universal Binary Neurons: Learning and Application to Image Processing", Proc. of the Third IEEE International Workshop on Cellular Neural Networks and their Applications; Italy; Roma: 1994; IEEE 94TH0693-2; 153-158.
- [5] Chua LO, Yang L. Cellular neural networks: Theory; IEEE Trans.Circuits Syst.; 1988; 35: 1257-1290.
- [6] Aizenberg IN. Universal Logical Element over the Field of the Complex Numbers; Kibernetika (Cybernetics); 1991; 3: 116-121.
- [7] Kohonen T. Content-Addressable Memories; Springer-Verlag; Berlin-Heidelberg-New-York: 1980.
- [8] Ramaher U, Raab W, Anlauf J, Hachmann U, Beichter J, Bruls N, Weseling M, Sichender E, Manner R, Glass J, Wurz A. Multi-processor and Memory Architecture of the Neurocomputer Synapse-1; Proceedings of the 3-d International Conference on Microelectronics for Neural Networks; UK; Edinburgh: 1993; 227-231.