

Агломеративный метод сегментации иероглифического текста

Кугаевских А.В.

Институт математики и компьютерных наук
Тюменский государственный университет, Тюмень, Россия
a-kugaevskikh@yandex.ru

Аннотация

В работе предложен новый алгоритм сегментации, основанный на агломеративном подходе. Основной принцип состоит в построении пирамиды сегментации, каждый уровень которой представлен в виде неориентированного графа. На первом уровне вершинами графа являются пиксели изображения, на втором и последующих – выделенные сегменты. Также строятся деревья вложенности сегментов. Верность сегментации проверяется подсистемой распознавания на каждом уровне, при этом сокращается количество непроверенных сегментов и корректируются деревья вложенности.

Ключевые слова: сегментация текста, агломеративная кластеризация, пирамида сегментации, деревья вложенности.

1. ВВЕДЕНИЕ

Сегментация текста, написанного по правилам древнеегипетской грамматики, необычайно трудная задача. Древнеегипетский текст содержит около 7000 символов, часто отличающихся только несколькими штрихами, и осложнен такими элементами как группировки и картуши, к тому же, часто практиковалось написание текста с разными ориентациями. Дополнительную сложность вносит написание текста в вертикальных колонках. Поэтому необходимо проводить анализ иерархической структуры документа и распознавать не только символы, но и грамматические элементы структуры текста (колонки, картуши и т.д.). Пример текста показан на рисунке 1.



Рисунок 1: Исходное изображение

Особым моментом является тот или иной уровень повреждения оригинала, что затрудняет как восстановление структуры текста, так и успешное распознавание, содержащих его, символов. В отличие от OCR-систем, работающих с используемыми на данный момент языками, в данном случае мы не можем привлечь на помощь лингвистику. Так как нашего знания древнеегипетского языка не достаточно для создания действительно работающего лингвистического модуля, который исправлял бы ошибки модуля распознавания, как это сделано, например, в Abbyy FineReader.

На данный момент разработано достаточно большое количество разнообразных подходов. Среди них нахождение границ через обнаружение разрывов яркости, выращивание областей, слияние и разделение регионов, метод морфологических водоразделов [1-2]. В последнее время активно развиваются методы, основанные на применении теории графов и оптимизационного подхода [3]. Общая идея методов, основанных на теории графов, следующая. Изображение представляется в виде взвешенного графа, с вершинами в точках изображения. Вес ребра графа отражает сходство точек в некотором смысле (расстояние между точками по некоторой метрике). Среди всех методов, основанных на теории графов, особо хочется отметить метод SWA [4-6]. Основная идея метода состоит в построении пирамиды взвешенных графов, каждый из которых получен из предыдущего путем объединения схожих вершин. На каждом шаге веса связей пересчитываются. В процессе построения пирамиды вычисляются различные статистики, характеризующие форму, цвет, текстуру регионов, эти статистики используются для вычисления меры сходства регионов. Затем, следуя идеологии методов теории графов, для полученного графа вводится функционал стоимости разреза и ищется разрез минимальной стоимости. На рисунке 2 показан результат применения метода SWA к изображению на рисунке 1 (слева) и правильная сегментация, выполненная вручную (справа). Как видно, сегментация изображения явно ошибочна.



Рисунок 2: Результат работы метода SWA (слева) и ручная сегментация (справа)

Подсистема сегментации должна формировать гипотезы о сегментах, содержащие предполагаемые границы символов или их элементов, и передавать их на подсистему распознавания для проверки правильности сегментирования.

2. ОБЩИЙ АЛГОРИТМ

Предлагаемый в данной работе метод является вариацией на тему агломеративной кластеризации пикселей [2]. Каждому пикселю ставится метка соответствия одному из классов (шум, фон, символ или его элемент). Для получения лучшей сегментации подобные в некоторой метрике пиксели рекурсивно сливаются. При этом предлагается многоуровневый алгоритм слияния токенов (заполнителей, указывающих на наличие в изображении искомой структуры), с проверкой правильности слияния на каждом уровне. На первом уровне токенами являются пиксели, являющиеся предполагаемыми граничными элементами сегментов, а на последующих – выделенные сегменты изображения.

Общий алгоритм сегментации таков:

1. Подготовительный этап
 - a. Для каждого пикселя изображения вычисляется уровень яркости
 - b. К матрице яркостей применяется фильтр Собеля для расчета уровня градиента
 - c. По матрице значений градиента вычисляется порог разделения фона и контуров объектов с помощью метода Отсу
2. Построение 1-ого уровня пирамиды сегментации
 - a. Построение неориентированного взвешенного графа по изображению
 - b. Агломеративная кластеризация пикселей
3. Построение последующих уровней пирамиды сегментации
 - a. Построение неориентированного взвешенного графа по набору сегментов
 - b. Построение деревьев вложенности сегментов
 - c. Агломеративная кластеризация сегментов

3. ПОДГОТОВИТЕЛЬНЫЙ ЭТАП

Методы поиска границ хорошо разработаны для полутоновых изображений. Полутоновое изображение рассматривается как функция двух переменных (x и y), и предполагается, что границы регионов соответствуют максимумам градиента этой функции. Для их поиска применяется аппарат дифференциальной геометрии (в простейшем случае это фильтры Робертса, Превитта, Собеля, лапласиан), позволяющий найти дискретное приближение двумерного градиента. Из приведенных фильтров, применение фильтра Собеля дает наименьший шум на изображении, поэтому именно он был выбран для вычисления градиента. На рисунке 3 показано применение фильтра Собеля к изображению текста.

Фильтр Собеля [7] основан на свёртывании изображения небольшими раздельными целочисленными фильтрами в вертикальном и горизонтальном направлениях. Фильтр использует ядра 3×3 , с которыми свёртывают оригинальное изображение к вычислению приближений производных по горизонтали и по вертикали.



Рисунок 3: Фильтр Собеля

Пусть A исходное изображение, а G_x и G_y — два изображения, где каждая точка содержит приближенные производные по x и по y . Они вычисляются следующим образом:

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A; G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} * A \quad (1)$$

где $*$ обозначает двумерную операцию свертки.

Координата x здесь определена возрастающей «направо», а y — «вниз». В каждой точке изображения полученное приближение градиента состоит из двух величин:

$$Gr = \sqrt{G_x^2 + G_y^2} \quad (2)$$

К матрице Gr применяется метод бинаризации Отсу [8] для получения порога разделения границы и фона. Алгоритм предполагает наличие в изображении двух классов пикселей и ищет оптимальный порог, разделяющий эти два класса так, что бы их внутриклассовая дисперсия была минимальна. Отсу показал, что минимизация девиации внутри класса это тоже самое, что и максимизация девиации между классами:

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = w_1(t)w_2(t)[\mu_1(t) - \mu_2(t)]^2 \quad (3)$$

которая выражается в терминах вероятности w_i и среднее арифметическое класса μ_i , которое в свою очередь может обновляться итеративно. Эта идея привела к эффективному алгоритму:

Строится гистограмма по значениям $p_i = n_i/N$, где N — это общее кол-во пикселей на изображении, n_i — это кол-во пикселей с уровнем градиента i . Диапазон градиентов делится на два класса с помощью порогового значения уровня градиента $t \in [0 \dots L]$. Каждому классу соответствуют относительные частоты:

$$w_1(t) = \sum_{i=1}^t p_i \quad (4)$$

$$w_2(t) = 1 - w_1(t) \quad (5)$$

Средние уровни для каждого из двух классов изображения:

$$\mu_1(t) = \sum_{i=1}^t \frac{ip_i}{w_1(t)} \quad (6)$$

$$\mu_2(t) = \sum_{i=t+1}^L \frac{ip_i}{w_2(t)} \quad (7)$$

Идея алгоритма состоит в максимизации межклассовой дисперсии $\sigma_b^2(t)$, вычисленной по формуле (3).

Таким образом, мы имеем две характеристики, отличающие соседние пиксели, это яркость пикселя и степень принадлежности к границе, выраженная значением градиента. Применение этих характеристик позволит нам построить контур сегмента.

4. ПОСТРОЕНИЕ ПИРАМИДЫ СЕГМЕНТАЦИИ

На первом уровне пирамиды сегментации исходное изображение представляется в виде неориентированного взвешенного графа $G_1 = (V, E)$, где вершинами V являются пиксели, соединенные друг с другом по принципу 8-смежности [1], т.е. со всеми пикселями-соседями, ребрами E , показывающими степень похожести пикселей. В качестве метрики похожести пикселей предложена формула (8), полученная эмпирическим путем:

$$w_{ij} = Gr_j |L_j - L_i| \quad (8)$$

где L_i - значение яркости текущего пикселя,

Gr_j - значение градиента соседнего пикселя,

L_j - значение яркости соседнего пикселя.

Для поиска сегмента применяется алгоритм Дейкстры, с той лишь разницей, что начальная и конечная вершины совпадают.

1. Ищется пиксель с максимальным значением градиента Gr
2. Вершина графа, соответствующая данному пикселю, становится начальной вершиной
3. Выполняется алгоритм Дейкстры для поиска замкнутого контура
4. Шаги 2-3 повторяются для всех оставшихся пикселей, значение градиента которых выше порога t , полученного по методу Отсу (формула 3).

Выход первого уровня характеризуется множеством найденных сегментов $S = [s_1, s_2, \dots, s_k]$. На рисунке 4 схематично показана часть сегментов, найденных на изображении с рисунка 1. Система строит сегменты по найденным контурам и, в идеале, они точно повторяют границы символа. Найденные области могут принадлежать символу, элементу символа или шуму. Они передаются на следующий уровень $l \in [2, \infty)$, который представляется в виде полного неориентированного взвешенного графа $G_l = (S, E)$, где вершинами являются найденные сегменты, соединенные друг с другом ребрами E , показывающими близость пары сегментов. Веса ребер являются w_{ij} - минимальное евклидово расстояние между точками токенов i и j .



Рисунок 4: Сегментация изображения

На втором уровне строятся также деревья вложенности сегментов, которые корректируются на последующих уровнях, рисунок 5. Деревья сегментов позволяют решить проблему с несколькими контурами одного и того же символа, как на рисунке 1, и играют главную роль при восстановлении структуры текста.

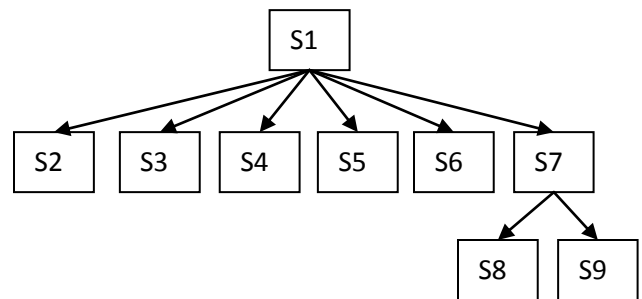


Рисунок 5: Дерево вложенности сегментов

На основании весов графов составляется список гипотез сегментации. Сегмент с наибольшим периметром становится опорным. Список гипотез составляется путем объединения токенов, при этом кандидаты на объединение берутся только при выполнении условия $w_{ij} \leq 3x$, где x - максимальное значение высоты или ширины токена. Каждая гипотеза об объединении токенов проверяется подсистемой распознавания. В результате для каждой гипотезы дается вероятность ее правдоподобия. Гипотеза, имеющая максимальную вероятность, признается истинной. Остальные токены передаются на следующий уровень, при этом множество S сокращается и корректируется дерево вложенности сегментов. Формирование уровней пирамиды и их проверка подсистемой распознавания продолжается до устранения неопределенности.

В случае невозможности классифицировать сегмент, возможно, имеет место недостаточная сегментация. При этом в список гипотез включается гипотеза о недостаточной

сегментации объекта. В этом случае сегмент делится на объекты, и к каждому из них применяется процедура распознавания. Предполагается, что точка сегментации лежит в области геометрического центра токена. В ней проводится линия, соответствующая наклону почерка сверху-вниз или справа-налево в зависимости от отношения вертикального и горизонтального размеров прямоугольника, описывающего токен. Полученные в результате распознавания данные сравниваются с остальными гипотезами. Процедура разбиения также носит иерархический характер и продолжается до устранения неопределенности.

Если символ распознан, но недостаточно уверенно (нормированный выход системы распознавания по этому классу от 50 до 70%), выделяется структура признаков распознанного символа. Каждый символ можно представить в виде неориентированного невзвешенного графа, элементами которого будут характерные линии, углы или пересечения линий, рисунок 6. Если в подобную структуру подставить данные по выходам подсистемы выделения признаков, можно проверить наличие опознанных признаков и, при необходимости, откорректировать гипотезы. Эти же признаки играют роль при обучении подсистемы распознавания. Таким образом, на основании данных распознавания признаков можно получить информацию об относительном расположении недостающих сегментов и, произведя их поиск на графе токенов, дополнить сегмент и увеличить вероятность верной сегментации и классификации.

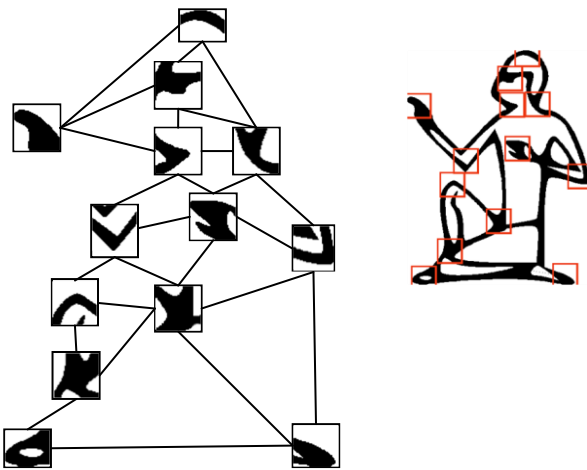


Рисунок 6: Структура признаков

5. ПРОВЕРКА СЕГМЕНТАЦИИ

Проверку гипотез можно возложить на формальные искусственные нейронные сети, хорошо зарекомендовавшие себя в решении задач распознавания образов. Из всего многообразия ИНС выгодно отличается неокогнитрон Фукушимы [9]. Данная сеть обладает требуемым уровнем инвариантности к сдвигу и деформации образов. Структура сети изображена на рисунке 7. Сеть делится на слои, слои на плоскости, плоскости на отдельные ячейки. Каждая плоскость слоя U_{C1} реагирует на изученные специфические признаки символа в определенном положении на изображении. Применительно к задаче сегментации нас интересуют выходы слоя U_{C1} (извлечение признаков) и U_{C4} (результат распознавания образа). Сеть обучается с помощью конкурентного контролируемого обучения. Учитель подает символ и указывает его признаки, отмечая характерные

линии, углы, пересечения линий. Для каждого нового признака должна быть создана новая плоскость в слое U_{S1} . Нейрон, оказавшийся в центре рецептивного поля и попадающий на признак, дает максимальный выход. Таким образом, обеспечивается конкурентное обучение всей плоскости. Нейроны последующих слоев имеют рецептивные поля большего размера. S-ячейки промежуточных стадий (U_{S2} и U_{S3}) самоорганизованы при помощи неконтролируемого конкурентного обучения. Уровень U_{C4} является уровнем распознавания, реакция которого показывает конечный результат распознавания образов сетью. Уровень U_{S4} обучается через контролируемое конкурентное обучение.

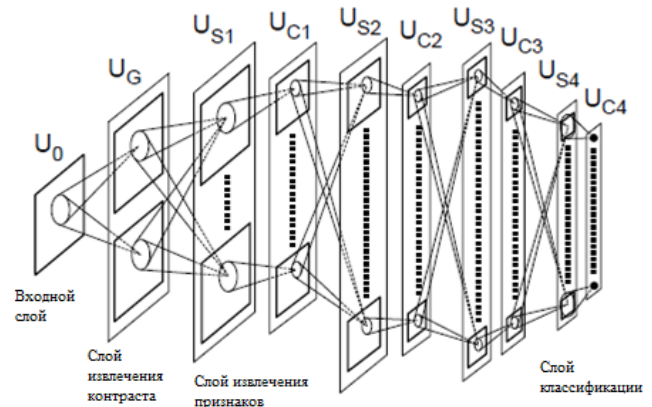


Рисунок 7: Структура неокогнитрона

Связи, ведущие к S-нейрону, имеют изменяемые весовые коэффициенты, получаемые в ходе обучения сети. В результате обучения S-нейрон активизируется только тогда, когда в его рецептивном поле оказывается именно тот признак образа, на который обучали данный нейрон. S-нейроны введены в сеть для того, чтобы уменьшить ошибку позиционирования. Каждый S-нейрон получает в качестве входа сигналы, поступающие от группы S-нейронов, которые выделяют одинаковый признак образа, но с небольшими смещениями в положении рецептивного поля. Функция активации S-нейрона такова, что S-нейрон переходит в активное состояние тогда, когда хотя бы один из этих S-нейронов находится в активном состоянии. Таким образом, S-нейрон распознает выделяемый элемент образа, даже если он немного сдвинут в своем положении, что обеспечивает инвариантность к сдвигам.

Каждый слой сети имеет более абстрактную, менее специфичную реакцию по сравнению с предшествующим; выходной слой реагирует на полные образы, показывая высокую степень независимости от их положения, размера и ориентации во входном поле. При этом для правильной работы классификатора, построенного на базе неокогнитрона, необходимо осуществлять предобработку изображения, подающегося на вход сети. У любой ИНС ограниченные размеры рецептивного поля, поэтому входное изображение необходимо масштабировать до необходимых размеров. С этой задачей очень хорошо справляется бикубическая интерполяция, обладая приемлемым качеством при высокой скорости работы алгоритма. Неокогнитрон налагает и свои требования: изображение должно быть бинарным, штрих символа должен обладать единичной толщиной. Бинаризовать изображение и получить штрих единичной толщины можно при векторизации изображения,

полученного путем слияния сегментов, находящихся на разных уровнях дерева вложенности. Внешний контур символа окажется на вышележащем уровне, тогда как внутренний контур – на нижележащем. Это изображение уже бинарно так как содержит только границы.

Примем выход фоторецепторных ячеек слоя U_0 за $u_0(n)$, где n показывает позицию ячейки. Выходом ячейки слоя U_G , где n - позиция центра рецептивного поля, является

$$u_G(n, k) = \varphi \left[-1^k \sum_{|v| < A_G} a_G(v) u_0(n + v) \right] \quad (9)$$

где $k=1,2$ и $\varphi [\]$ определяется как $\varphi [x] = \max(x; 0)$. Параметр $a_G(\xi)$ показывает силу фиксированной связи с ячейкой. Слой U_G состоит из двух плоскостей: одна содержит ячейки с оп-центром ($k=2$), другая ячейки с off-центром ($k=1$). A_G обозначает радиус диапазона суммирования v , т.е., размера пространственного распространения входных связей с ячейкой, что суть есть радиус рецептивного поля нейрона.

Входные связи с единственной ячейкой слоя U_G проектируются таким образом, чтобы их итоговая сумма была равна 0. Другими словами, связь $a_G(\xi)$ должна удовлетворять

$$\sum_{|v| < A_G} a_G(v) = 0 \quad (10)$$

где v - индекс связи, k - номер плоскости, n - индекс ячейки.

Для расчета массива параметров a_G чаще всего применяются вейвлеты.

Пусть $u_{S1}(n; k)$ и $u_{C1}(n; k)$ являются выходами S- и C-ячеек k -ой плоскости стадии 1, где n представляет позицию центра рецептивного поля ячейки, v - индекс связи, k - номер плоскости. Слой U_{S1} содержит также V-ячейки, выход которых определяется как $v_1(n)$. Выход S- и V-ячеек соответственно равен:

$$u_{S1}(n, k) = \frac{\theta_l}{1 - \theta_l} \varphi \left[\frac{1 + \sum_{\kappa=1}^{K_{Cl-1}} \sum_{|v| < A_{S1}} a_{S1}(v, \kappa, k) u_{Cl-1}(n + v, \kappa)}{1 + \theta_l b_{S1}(k) v_1(n)} - 1 \right] \quad (11)$$

$$v_1(n) = \sqrt{\sum_{\kappa=1}^{K_{Cl-1}} \sum_{|v| < A_{S1}} c_{S1}(v) \{u_{Cl-1}(n + v, \kappa)\}^2} \quad (12)$$

Параметр $a_{S1}(v, \kappa, k) (\geq 0)$ - сила переменной возбуждающей связи, идущей от C-ячейки $u_{Cl-1}(n + v, \kappa)$ предыдущей стадии. Следует отметить, что все ячейки плоскости совместно используют то же самое множество входных связей, следовательно $a_1(v, \kappa, k)$ не зависит от n . A_{S1} обозначает радиус рецептивного поля ячейки, т.е., размер пространственного сдвига входящего образа относительно центра рецептивного поля. Параметр $b_1(k) (\geq 0)$ - сила переменной тормозящей связи, идущей от V-ячейки. Параметр $c_{S1}(v)$ представляет силу постоянной возбуждающей связи с V-ячейкой, и является монотонно убывающей функцией от $|v|$. В данной работе он вычисляется также с помощью вейвлета «сомбреро», но берутся только положительные значения.

Положительная константа $\theta_l (< 1)$ является порогом чувствительности S-ячейки и определяет степень селективности в извлечении признаков.

В (3) и (4) для $l = 1$, $u_{Cl-1}(n; k)$ заменяется на $u_C(n; k)$, и мы имеем $K_{Cl-1} = 2$, где K - количество плоскостей.

Каждой S-ячейке соответствует своя V-ячейка.

Схематично связи между ячейками можно представить таким образом как они отображены на рисунке 8.

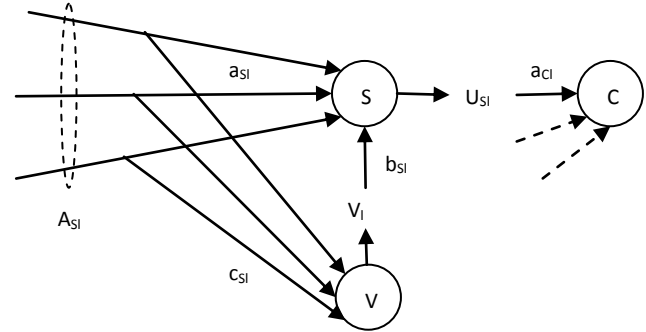


Рисунок 8: Схема связей между ячейками в слое

Задачей комплексных ячеек является уменьшение зависимости реакции системы от позиции образов во входном поле. Для достижения этого каждый комплексный узел получает в качестве входного образа выходы набора простых узлов из соответствующей плоскости того же слоя. Эти простые узлы покрывают непрерывную область простой плоскости, называемую рецептивной областью комплексного узла. Возбуждение любого простого узла в этой области является достаточным для возбуждения данного комплексного узла. Таким образом, комплексный узел реагирует на тот же образ, что и простые узлы в соответствующей ему плоскости, но он менее чувствителен к позиции образа, чем любой из них.

Таким образом, каждый слой комплексных узлов реагирует на более широкую область входного образа, чем это делалось в предшествующих слоях. Эта прогрессия возрастает линейно от слоя к слою, приводя к требуемому уменьшению позиционной чувствительности системы в целом.

Ответ S-ячейки уровня U_{S1} пространственно размыт в последующих C-ячейках уровня U_{C1} . Математическим ответом S-ячейки U_{C1} , исключая самую высокую стадию U_{C4} , является

$$u_{C1}(n, k) = \psi \left[\sum_{|v| < A_{C1}} a_{C1}(v) u_{S1-1}(n + v, \kappa) \right] \quad (13)$$

где $\psi [x] = \frac{\varphi [x]}{1 + \varphi [x]}$. Параметр $a_{C1}(v)$ показывает силу постоянных возбуждающих связей, сходящихся от группы S-ячеек, которые распространяются в пределах радиуса A_{C1} .

Плоскость C-ячеек соединена только с соответствующей плоскостью S-ячеек.

Обучение сети выполняется от более низких стадий к более высоким: после завершения обучения более низкой стадии, начинается обучение следующей. То же самое множество обучающих образов используется для обучения всех стадий, кроме U_{S1} .

Хотя метод выбора начальных параметров ячеек во время обучения немного отличается в разных слоях, правило усиления изменяющихся связей $a_1(v, \kappa, k)$ и $b_1(k)$ одинаково для всех слоев.

Сначала выбирается простой узел, веса синапсов которого должны быть увеличены, он рассматривается как представитель всех узлов в плоскости, вызывая увеличение их связей на том же самом образе. Это делается каждый раз, когда представляется новый обучающий образ и для каждой плоскости. Представители выбираются из тех S-ячеек, которые дают большой выход, но число представителей ограничивается так, чтобы был не более чем один представитель от любой одной S-плоскости. Таким образом, все узлы в плоскости обучаются распознавать одни и те же свойства, и после обучения будут делать это независимо от позиции образа в поле комплексных узлов в предшествующем слое.

Для представителя S-ячейки усиливаются только входные соединения, через которые приходят не равные нулю сигналы. С этой процедурой представитель S-ячейки становится выборочно отзывчивым только на те характеристики стимула, который представлен. Все другие S-ячейки в S-плоскости, от которых выбран представитель, имеют такое же усиление своих входных соединений, как и их представитель. Эти отношения могут количественно быть выражены следующим образом.

Пусть ячейка $u_{SI}(\hat{n}, \hat{k})$ была выбрана в качестве начальной, следовательно, изменяющиеся связи $a_l(v, \kappa, \hat{k})$ с этой ячейкой, как и со всеми S-ячейками в этой плоскости, увеличиваются следующим образом:

$$\Delta a_l(v, \kappa, \hat{k}) = q_l c_{SI}(v) u_{CI-1}(\hat{n} + v, \kappa) \quad (14)$$

где q_l - положительная константа, определяющая скорость обучения. Хотя было предложено много алгоритмов определения тормозящей связи $b_l(\hat{k})$, мы используем метод, в котором $b_l(\hat{k})$ определяется исходя непосредственно из возбуждающих связей $a_l(v, \kappa, \hat{k})$, следующим образом:

$$b_{SI}(\hat{k}) = \sqrt{\sum_{\kappa=1}^{K_{CI-1}} \sum_{|v| < A_{SI}} \frac{\{a_{SI}(v, \kappa, \hat{k})\}^2}{c_{SI}(v)}} \quad (15)$$

Однако в ячейках S-плоскости, от которых не выбран представитель, входные соединения не изменяются. Выбор начальных величин изменяемых соединений имеет небольшой эффект для надежности неокогнитрона, при условии, что они невелики и определены таким способом, что каждая S-плоскость имеет другой ряд начальных величин для своих входных соединений.

S-ячейки промежуточных уровней (U_{S2} и U_{S3}) самоорганизованы при помощи неконтролируемого конкурентного обучения, применяемого в стандартном неокогнитроне. Начальные ячейки определяются принципом "Победитель получает все" (WTA). Каждый раз, когда обучающий образ подается на входной слой, каждая S-ячейка конкурирует с другими ячейками поблизости в области, называемой областью конкуренции. Если выход ячейки больше остальных в области конкуренции, ячейка выбирается в качестве начальной, при этом в плоскости может и не быть выбран ни один представитель. Как можно увидеть по формуле (14), каждая входная связь начальной ячейки увеличивается на число пропорциональное ответу ячейки, от которой идет связь. Из-за совместного использования связей в пределах каждой плоскости, все ячейки плоскости имеют то же множество входных связей, что и начальная ячейка.

Так как представители определяются этим способом, каждая S-плоскость становится выборочно чувствительной на одну из характеристик обучающих образов и нет возможности образования излишних соединений так, чтобы две или более S-плоскостей использовались для обнаружения одной и той же характеристики. Случайно представители выбираются только из небольшого количества S-плоскостей за один раз, остальная часть S-плоскостей производит представителей, когда представляются другие стимулирующие образцы. S-ячейки самой высокой стадии (U_{S4}) обучаются при помощи контролируемого конкурентного обучения. Правило обучения напоминает конкурентное обучение, использованное в слоях U_{S2} и U_{S3} , однако здесь также используются имена классов обучающих образов. Когда сеть изучает варианты деформированных обучающих образов через конкурентное обучение, более чем одна плоскость для одного класса обычно создается в U_{S4} . Поэтому каждая плоскость изучает обучающий образ и имя класса.

По результатам моделирования и серии экспериментов получились следующие параметры сети:

1. Размер слоя U_0 : 65*65
2. Размер плоскости слоя U_G : 71*71, a_G 7*7, $A_G=3.3$, $A_{g0}=1.2$
3. Размер плоскости слоя U_{S1} : 68*68, количество плоскостей 47915, a_{S1} 6*6, $\theta_1 = 0.55$
4. Размер плоскости слоя U_{C1} : 37*37, количество плоскостей 47915, a_{C1} 18*18
5. Размер плоскости слоя U_{S2} : 38*38, количество плоскостей 95850, a_{S2} 6*6, $\theta_2 = 0.51$
6. Размер плоскости слоя U_{C2} : 21*21, количество плоскостей 95850, a_{C2} 14*14
7. Размер плоскости слоя U_{S3} : 22*22, количество плоскостей 143725, a_{S3} 6*6, $\theta_3 = 0.58$
8. Размер плоскости слоя U_{C3} : 13*13, количество плоскостей 143725, a_{C3} 8*8, $A_{C3}=5$
9. Размер плоскости слоя U_{S4} : 5*5, количество плоскостей 142785, a_{S4} 9*9, $\theta_4 = 0.30$
10. Размер плоскости слоя U_{C4} : 1*1, количество плоскостей 6845, a_{C4} 5*5

На обучающей выборке из 6845 символов сеть показала качество распознавания в 99%, на тестовой выборке из 27380 символов, представляющих те же самые обучающие образы с повреждениями в каждой из четвертей символа, сеть показала результат в 86% [10]. Тестовая выборка создавалась путем искажения символов из обучающей выборки в каждой из четвертей квадрата символа.

6. ЗАКЛЮЧЕНИЕ

Построение пирамиды и проверка правильности сегментации на каждом уровне позволяет сократить количество ошибочно выделенных сегментов и, как следствие, повысить качество распознавания символов или элементов текста. Деревья вложенности сегментов не только необходимы для восстановления структуры текста, но и, участвуя в распознавании, также позволяют повысить его качество, дополняя информацию о сегменте. Обратные связи от подсистемы распознавания к подсистеме сегментации позволяют сократить количество проверяемых гипотез и улучшить качество сегментации.

7. СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Гонсалес Р., Вудс Р. Цифровая обработка изображений.- М.: Техносфера, 2005
2. Форсайт Д., Понс Ж. Компьютерное зрение. Современный подход.: Пер. с англ.- М.: Издательский дом «Вильямс», 2004
3. Вежнев А., Барина О. Методы сегментации изображений: автоматическая сегментация. Компьютерная графика и мультимедиа, 4(4), 2006. <http://cgm.computergraphics.ru/content/view/147>
4. Sharon E., Brandt A., Basriy R. Fast Multiscale Image Segmentation. CVPR 2000
5. Sharon E., Brandt A., Basriy R. Segmentation and Boundary Detection Using Multiscale Intensity Measurements. CVPR 2001
6. Galun M., Sharon E., Brandt A., Basriy R. Texture Segmentation by Multiscale Aggregation of Filter Responses and Shape Elements. ICCV 2003
7. Дуда Р., Харт. П. Распознавание образов и анализ сцен. Пер. с англ. – М.: Мир, 1976.
8. Otsu N. A threshold selection method from gray-level histograms. IEEE Trans. Sys., Man., Cyber. 9: 62-66, 1979.
9. Fukushima K. Neocognitron for handwritten digit recognition, Neurocomputing, 51, pp. 161-180, 2003.
10. Кугаевских А.В., Ивашко А.Г. Возможность применения искусственных нейронных сетей "неокогнитрон" для распознавания древнеегипетских иероглифов // Вестник Тюменского Государственного Университета, 2009, №6, с. 209-214.

Об авторе

Кугаевских Александр Владимирович, аспирант Тюменского государственного университета, факультет математики и компьютерных наук. Контактный e-mail: a-kugaevskikh@yandex.ru.