

УДК 0.04.93'1 + 004.383.8

ТЕОРИЯ РАСПОЗНАВАНИЯ ОБРАЗОВ И НЕЙРОКОМПЬЮТЕРЫ

М.М. ТАТУР

Белорусский государственный университет информатики и радиоэлектроники
П. Бровки, 6, Минск, 220027, Беларусь

Поступила в редакцию 27 января 2014

Представлен новый подход к применению ключевых положений теории распознавания образов при построении параллельных проблемно-ориентированных процессоров. Задачи классификации кластеризации и ассоциирования рассмотрены с позиций унификации параллельной архитектуры процессора. В качестве конечного результата анонсирован прототип отечественного нейроподобного компьютера.

Ключевые слова: класс, кластер, ассоциация, параллельные вычисления, SIMD-архитектура, нейрокомпьютер.

Введение

При решении прикладных задач построения интеллектуальных систем научное направление, известное как распознавание образов, часто и, порой, неразрывно, соседствует с родственными направлениями – теорией оптимизации принятия решений (исследование операций), теорией обработки сигналов и изображений, теорией семантической обработки знаний (искусственный интеллект). Очень часто задачи интеллектуального принятия решений требуют специальных средств вычислений, поскольку их включают в контуры управляющих систем реального времени, а переборный, комбинаторный характер вычислений обуславливает нелинейное возрастание сложности с ростом числа переменных. Поэтому распознавание образов и связанные с ним направления органично сочетаются с распараллеливанием вычислений и реализацией на суперЭВМ, вычислительных кластерах и проблемно-ориентированных компьютерах (процессорах) с оригинальными архитектурами. Состав и связь указанных научных направлений отражены на рис. 1.

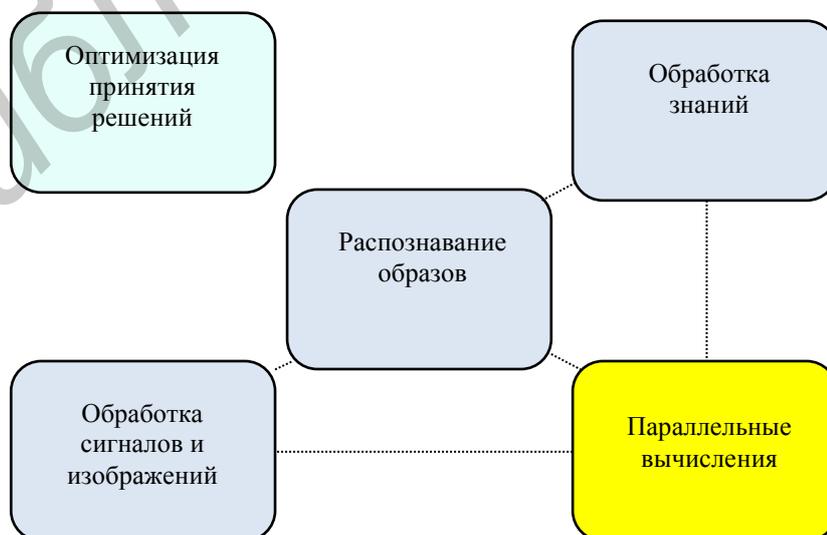


Рис. 1. Научные направления, связанные с проблемой построения интеллектуальных систем

В МРТИ–БГУИР на различных кафедрах, в различных научных школах развивалась дисциплина «Цифровая обработка сигналов», как в классическом варианте, так и применительно к прикладным задачам. С приходом из Института технической кибернетики (ныне Объединенный институт проблем информатики – ОИПИ НАН Беларуси) д.т.н., профессора Голенкова В.В. (зав. кафедрой интеллектуальных информационных технологий с 1995 г.) в университете стало активно развиваться направление обработки знаний. В 1995 г. с того же института перешел на работу в БГУИР и возглавил кафедру ЭВМ д.т.н., профессор Садыхов Р.Х. Его научная деятельность сосредоточилась в направлениях обработки изображений и распознавания образов, а результативность претендует на признание научной школы [1–3].

Его ученики – д.т.н., профессор Головки В.А. (зав. кафедрой Интеллектуальных информационных технологий Брестского государственного технического университета), д.т.н. Дудкин А.А. (главный научный сотрудник ОИПИ НАН Беларуси), д.т.н. Татур М.М. (профессор кафедры ЭВМ БГУИР) в своих работах развивают и преумножают идеи своего наставника [4–31].

Когнитивный (интеллектуальный в полном смысле) процесс можно рассматривать как многоуровневый (иерархический) итерационный (динамический) процесс синтеза и анализа информации, включающий формирование образов, определение классов образов, распознавание образов, оценку качества распознавания, уточнение образов, уточнение классов и т.д. Ключевыми составляющими такого процесса являются формальные задачи: кластеризация, классификация и ассоциирование, которые находят вполне конкретные приложения в прикладных системах интеллектуальной обработки данных.

В настоящей работе приводится оригинальная трактовка этих понятий как взаимодополняющих частей интеллектуального анализа данных.

Основные понятия и определения

Если определить класс как идентификатор (метку), который присваивается образу, удовлетворяющему некоторому формальному правилу (критерию). Тогда классификация – это процесс отнесения неизвестного образа к одному из определенных классов. В качестве правила классификации может выступать заданная мера близости (подобия) либо область в признаковом пространстве, в рамках которой все образы считаются однородными (одинаковыми, неразличимыми).

Если кластер – это совокупность однородных в определенном смысле образов, тогда кластеризация – процесс группировки (структурирования) образов по заданному критерию однородности.

Если ассоциация – это некая связь (напр. общность) между классами посредством некоторого образа, который может принадлежать различным классам, тогда ассоциирование – это процесс поиска возможности выхода образа за рамки одного класса и отнесения к различным классам или, другими словами, поиск того, что объединяет разные классы.

Понятие однородности или различия образов является фундаментальным, через которое определяются основные понятия – класс, кластер и ассоциация. Для количественной оценки степени однородности используются различные меры (метрики): евклидово расстояние, расстояние Хэмминга, расстояние Танимото, метрика Минковского и др. Приведенная система терминов не является общепринятой и отражена мнемосхемой на рис. 2. Число информативных признаков с целью достижения наглядности примера ограничено двумя.

Обучение в широком смысле – это неотъемлемый процесс настройки (адаптации) интеллектуальной системы, направленный на повышение эффективности анализа данных. В зависимости от вида интеллектуального анализа (классификация, кластеризация, ассоциирование), а также от конкретного метода реализации, обучение может приобретать совершенно различные формы и трактовки.

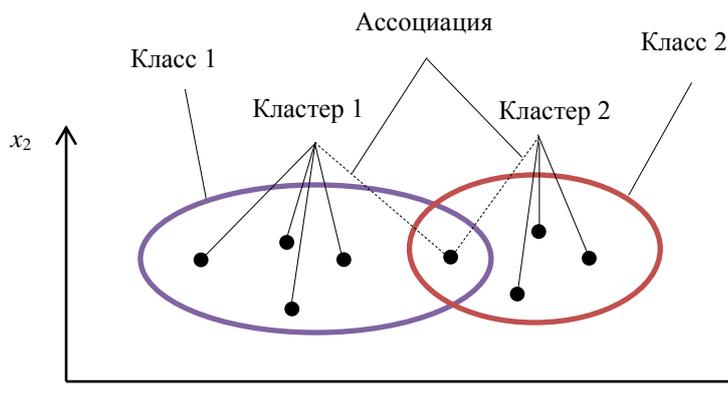


Рис. 2. Иллюстрация понятий «класс-кластер-ассоциация», которые отражают различные аспекты интеллектуального анализа данных

Формальная задача классификации

Пусть $X: (x_1, x_2, \dots, x_n)$ – вектор (или образ), заданный значениями x_i в n -мерном пространстве информативных признаков, $Y: (y_1, y_2, \dots, y_k)$ – множество идентификаторов (меток) классов. Известна обучающая выборка, представленная парами значений «образ-класс» $X \times Y$ на конечном множестве образов, при этом функциональная зависимость Ψ , которая относит образ к одному из k -классов, неизвестна:

$$Y = \Psi(X). \quad (1)$$

Тогда, задача построения классификатора состоит в том, чтобы найти (подобрать) функцию f такую, чтобы выходные значения $Y: (y_1, y_2, \dots, y_k)$ имели минимальную, например среднеквадратическую, ошибку на тестовой выборке.

Обычно функционал f является общим для всех классов, а подстройка под каждый класс осуществляется за счет настроек U_j ($j = 1-k$). Общая схема классификации, для случая k -классов, приведена на рис. 3.

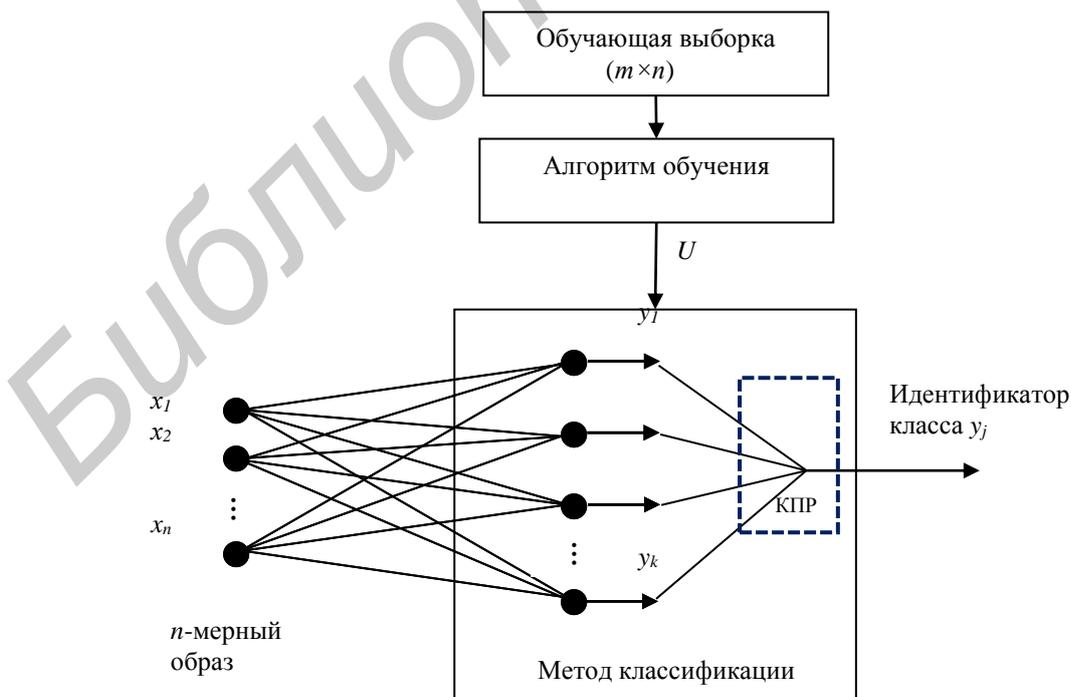


Рис. 3. Общая схема классификации

Таким образом, основной целью классификатора является достижение максимального соответствия (равенства) (2), поэтому часто классификаторы называют аппроксиматорами.

$$\Psi(X) \approx f(X, U). \quad (2)$$

В действительности, перечень возможных функций f известен, предопределен из опыта развития данного научного направления и описан в литературе как математические методы классификации. Наиболее популярные методы классификации, среди которых: группа методов, основанных на оценке расстояний между образами, группа методов, основанных на построении разделяющих поверхностей (дискриминантных функциях) и группа методов, основанных на элементарных знаниях (логических зависимостях) [23].

В принципе, каждый информативный признак каким-либо образом может оказывать влияние на принятие решения, что и отображено на граф-схеме в виде полносвязного графа признаки-классы: $k \times n$. Непосредственное, финальное решение определяется по критерию принятия решения (КПР), который в конечном итоге сводится к выбору оптимального уровня порога для четкого способа принятия решения, либо параметров функции нечеткого принятия решения.

При классификации могут быть использованы две альтернативные модели принятия решений – четкая и нечеткая (рис. 4). При четком принятии решения определяется порог p , с которым сравнивается вычисленное значение $f(X, U)$. Очевидно, что получать решения с «грубыми» данными не представляет особого труда, однако, наиболее «тонкие» решения приходится принимать в непосредственной близости к порогу. Очень часто в реальных системах присутствуют шумы и неопределенности, которые не позволяют в окрестности порога однозначно разделить решения на «положительные» и «отрицательные», при этом резко возрастает вероятность ошибок в принятии решений.

Для подобных задач математический аппарат разработан достаточно полно и позволяет легко получать количественные оценки достоверности, рассчитывать и оптимизировать ошибки первого и второго родов, возможные потери и т.д. (например, аппарат вероятностной оценки принятия решения по стратегии Байеса. Смещением порога можно целенаправленно управлять вероятностью появления ошибок первого либо второго родов (перераспределять вероятности ошибок), однако исключить их невозможно.

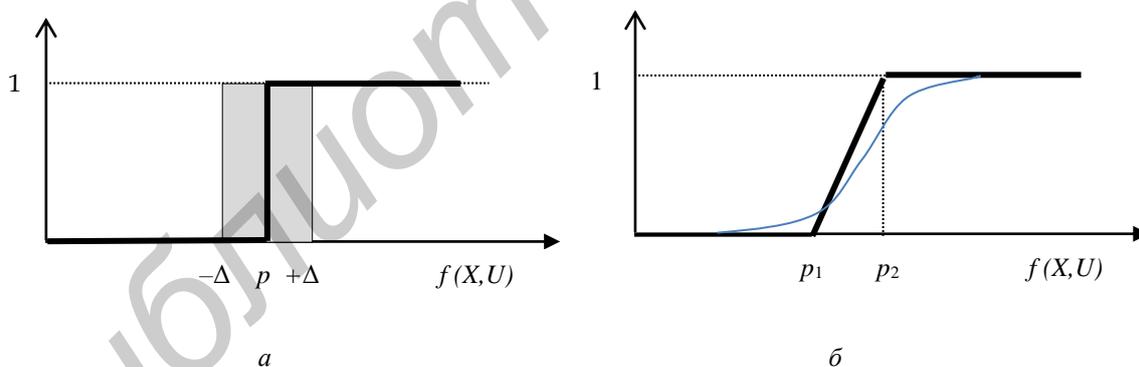


Рис. 4. Модели способов четкого (а) и нечеткого (б) принятия решений

В отдельных прикладных задачах, когда наличие ошибок в системе недопустимо, в окрестности порога ($\pm\Delta$) вводят подкласс выбросов, т.е. отказ от принятия решения. В таких случаях необходимо решать оптимизационную задачу выбора «коридора порогов».

Итак, в системах с четким принятием решений величина возможной ошибки – максимальна и дискретна. Однако, в некоторых задачах распознавания и управления допускается и даже желательно применение нечеткого (размытого) решения вблизи порога. Тогда используют нечеткий (fuzzy) способ принятия решений, который может быть реализован как в виде непрерывной, так и кусочно-линейной функцией с порогами p_1 и p_2 , рис. 4, б. Тогда на выходе системы будет сформирована величина степени принадлежности в интервале $[0, 1]$, причем возможные «неправильные» значения выходной функции в окрестности порога будут иметь относительно небольшие отклонения от «правильных».

Формальная задача кластеризации

Пусть $X_j: (x_{1j}, x_{2j}, \dots, x_{nj})$ – вектор (или образ), заданный значениями x_i в n -мерном пространстве информативных признаков. Всего имеется p таких образов. В качестве параметров кластеризации могут быть заданы: число кластеров, степень их сходства или различия, мера сходства образов и др. В результате вычисления одним из методов кластеризации все множество образов будет разделено на группы, т.е. каждому образу будет присвоена метка класса.

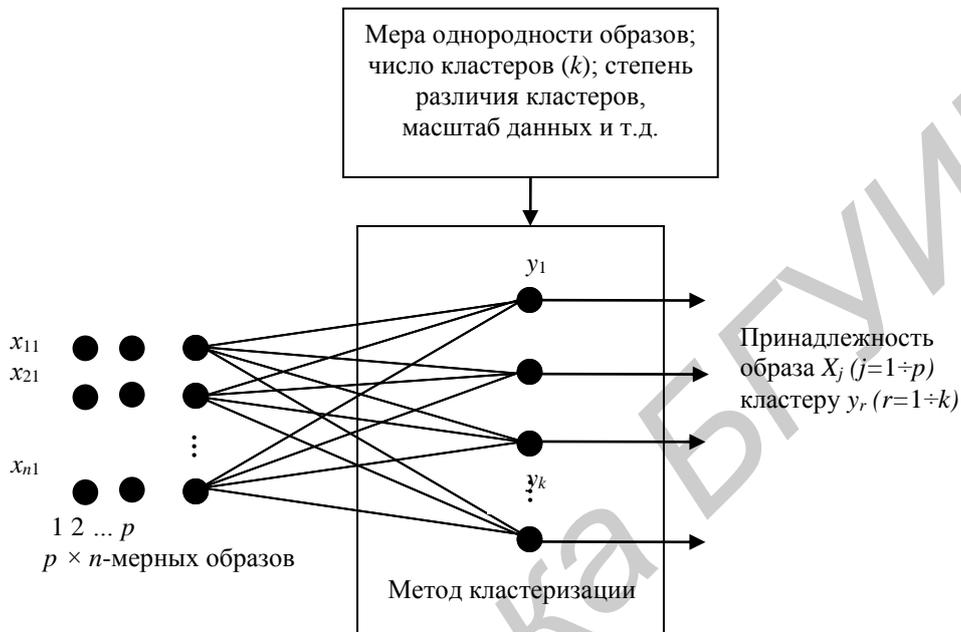


Рис. 5. Общая схема кластеризации

Задача кластеризации, по сути, есть оптимизационная задача, в которой минимизируется различие (расстояние) между образами одного кластера, и максимизируется различие между образами различных кластеров. Наиболее яркими представителями методов кластеризации являются иерархические методы, основанные на полном переборе расстояний между образами, и итерационные (метод k - средних, и его разновидности). Как в задачах классификации, так и в задачах кластеризации существуют четкие и нечеткие варианты решений.

Формальная задача поиска ассоциативных правил

Целью поиска ассоциативных правил является нахождение закономерностей между связанными событиями. Ассоциативное правило имеет вид «Из события А следует событие В». Применительно к терминологии настоящей статьи в качестве «события» будем рассматривать отнесение образа X_i к некоторому классу, а связь между событиями – связь между классами, в данном случае между 1 и 2 (рис. 2).

Очевидно, что связь будет более существенной, если классы перекрываются в большей степени. Такое «перекрытие» можно регулировать изменением критерия принятия решений (рис. 3), либо сокращением числа информативных признаков. Для получения количественных показателей, отражающих связь классов между собой, можно ввести интуитивно понятные характеристики, наподобие известным – поддержке и достоверности. В целом, общая схема ассоциации будет такой же, как и для классификации, с той лишь разницей, что будут выдаваться несколько идентификаторов классов. При этом, сохраняется актуальность использования такой схемы в прикладных когнитивных задачах.

Архитектура нейроподобного компьютера

Как видно из рис. 3 и 5, общие схемы классификации, кластеризации и ассоциирования в своей основе имеют одинаковый вычислительный граф. Это позволяет работать над созданием общей модели параллельной архитектуры для различных когнитивных задач. В целом, такая архитектура относится к SIMD-типу параллельных процессоров (Single Instruction Stream – Multiple Data Stream), что означает один поток команд и множество потоков данных. В таком процессоре все процессорные элементы (ПЭ) – идентичны и работают с локальной памятью (рис. 6).

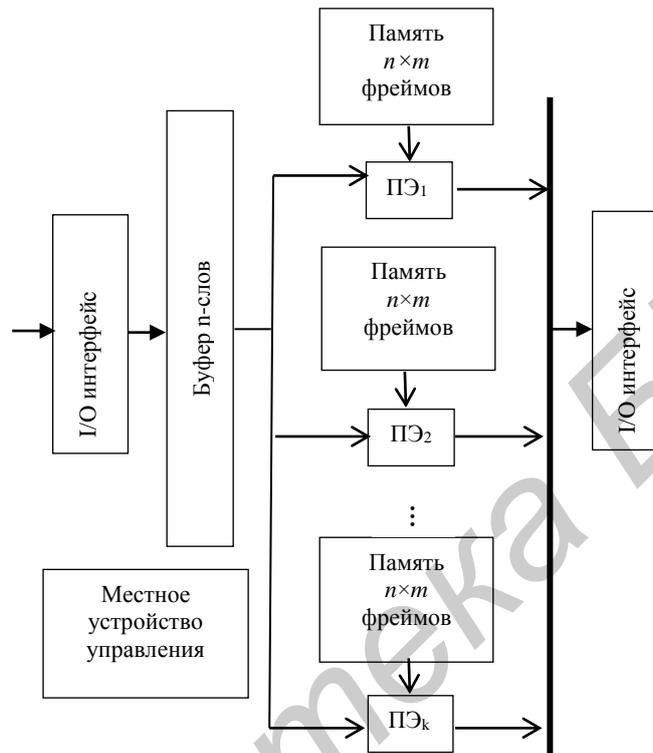


Рис. 6. Архитектура параллельного SIMD-процессора

Каждый ПЭ может быть настроен на аппаратную реализацию одной из когнитивных задач в рамках определенной математической модели [25, 26]. В данной модели применена нечеткая параметризация информативных признаков и реализована функция логических условий, что обеспечивает вычисление функций расстояния либо формирование нелинейных разделяющих поверхностей.

Для апробации теоретических заключений был разработан и изготовлен экспериментальный образец SIMD-процессора на базе FPGA Xilinx Spartan 6 со следующими выходными характеристиками: 8 ПЭ с объемом фрейма памяти – 2 кБ и временем вычисления по одному признаку – 50 нс. Общее время классификации (T), непосредственно характеризующее производительность процессора, определяется из выражения $T \approx (50nk)/q$.

Процессор представляет собой унифицированное программно-настраиваемое вычислительное устройство, позволяющее решать различные задачи классификации, кластеризации, ассоциации в диапазоне параметров n и k , определяемом емкостью фрейма памяти, в данном случае соотношением: $n \times k = 100$. При необходимости расширить функциональные возможности процессора достаточно либо использовать более мощную (ресурсоемкую) FPGA, либо разместить проект в нескольких FPGA, при минимальных издержках на архитектурное и структурное перепроектирование.

Очевидно, что для исследования функциональных возможностей процессора и проведения натурных испытаний необходимо формировать поток входных данных, принимать результаты промежуточных вычислений, управлять критериями принятия решений и выполнять ряд других сервисных функций. С этой целью была создана экспериментальная

вычислительная система (рис. 7), в рамках которой функции ввода-вывода данных, обучения и конечного принятия решений возлагаются на универсальный компьютер, а массовые функции классификации (кластеризации, ассоциирования) – на проблемно-ориентированный сопроцессор. Host-компьютер и сопроцессор обмениваются командами и данными по стандартным интерфейсам (USB). Такое архитектурное решение позволяет сконцентрировать усилия на исследовании программно-аппаратной реализации нейромодуля, при этом реализована возможность удобно обеспечивать общее управление, пользовательский интерфейс и при необходимости предобработку и постобработку обычными программными средствами [27, 28].

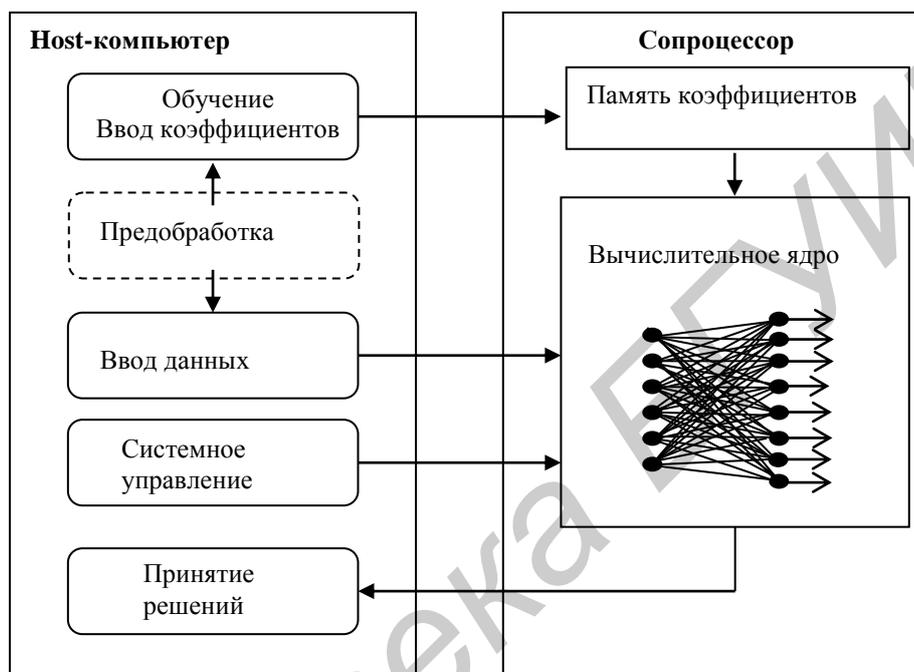


Рис. 7. Структурная схема нейроподобного компьютера

Заключение

Разработанный нейроподобный компьютер представляет собой инструментальное средство, которое позволяет создавать и исследовать широкий класс систем распознавания. Параллельный процессор может быть быстро и с минимальными затратами адаптирован к конкретной прикладной интеллектуальной системе, к конкретным конструктивным особенностям (в виде автономного блока либо сопроцессора).

Настоящая разработка явилась основой инновационного проекта «Отечественные нейроподобные компьютеры с массовым параллелизмом для решения задач идентификации в медицине, криминалистике, радиолокации, машиностроении, в научных исследованиях и образовании», осуществляемого Белорусским государственным университетом информатики и радиоэлектроники, Государственным предприятием «Научно-технологический парк БНТУ «Политехник» и ООО «Интеллектуальные процессоры». Данный проект занял второе место в конкурсе на «Лучший инновационный проект 2010 г.», проводимый Государственным комитетом по науке и технологиям, стал лауреатом конкурса Лучший инновационный проект и лучшая научно-техническая разработка года, проводимого 15–17 марта 2011 г. в Санкт-Петербурге. Экспериментальный образец изделия «Нейронный компьютер с наращиваемой программируемой архитектурой «Радимич-1» внедрен в производственном процессе Государственного предприятия «Научно-технологический парк БНТУ «Политехник». Начиная с 2011 г. изделие предлагается на рынке высоких технологий как в базовой конфигурации, так и под заказ. Опыт, полученный при создании проблемно-ориентированного процессора, передается студентам Белорусского государственного университета информатики и радиоэлектроники при чтении соответствующих курсов.

PATTERN RECOGNITION THEORY AND NEUROCOMPUTERS

M.M. TATUR

Abstract

The new approach to use of key statements of pattern recognition theory for the parallel problem oriented processors design is presented. The tasks of classification, clusterization and association are considered from point of view of parallel processor architecture unification. As a final result the domestic neuron like computer prototype is announced.

Список литературы

1. *Sadykhov R.H., Tatur M.M.* // Advances in modeling & analysis. AMSE Per. 1995. Vol. 30, № 3. P.23–30.
2. *Садыхов Р.Х., Татур М.М.* // Весці НАН Беларусі. Сер. фіз.-тэх. навук. 1995. № 3. С. 54–58.
3. *Садыхов Р.Х., Татур М.М., Одинец Д.Н.* // Весці НАН Беларусі. Сер. фіз.-тэх. навук. 1999. № 2. С. 79–83.
4. *Татур М.М., Садыхов Р.Х.* // Автоматика и вычислительная техника. 1997. № 1. С. 71–76.
5. *Татур, М.М., Садыхов Р.Х.* // Автоматика и вычислительная техника. 1998. № 3. С. 43–49.
6. *Татур М.М., Садыхов Р.Х.* // Микроэлектроника. 2000. № 2. С. 143–148.
7. *Садыхов Р.Х., Татур М.М.* Технический сервис однородных вычислительных устройств. Минск, 2001.
8. *Татур М.М., Одинец Д.Н.* Классификаторы в системах распознавания: прикладные аспекты синтеза и анализа. Минск, 2008.
9. *Алькоффаи М.С., Татур М.М., Садыхов Р.Х.* // Весці НАН Беларусі. Сер. фіз.-тэх. навук. 2005. №2. С. 75–80.
10. *Татур М.М.* //Автоматика и вычислительная техника. 1991. № 4. С. 49–54.
11. *Татур М.М.* // Микроэлектроника. 1992. №5. С. 32–36.
12. *Татур М.М.* //Автоматика и вычислительная техника. 1994. № 4. С. 76–78.
13. *Татур М.М.* //Микроэлектроника. 1997. № 1. С. 52–56.
14. *Алькоффаи М.С., Татур М.М.* // Докл. БГУИР. 2003. № 4. С. 86–87.
15. *Алькоффаи М.С., Татур М.М.* // Весці НАН Беларусі. Сер. фіз.-тэх.навук. 2004. № 1. С. 87–93.
16. *Байрак С.А., Татур М.М.* // Искусственный интеллект. 2006. № 3. С. 571–574.
17. *Калабухов Е.В., Садыхов Р.Х., Татур М.М.* // Весці НАН Беларусі. Сер. фіз.-тэх. навук. 2005. № 4. С. 96–99.
18. *Одинец Д.Н., Татур М.М.* // Автоматика и вычислительная техника. 1998. № 3. С. 31–37.
19. *Прытков В.А., Садыхов Р.Х., Татур М.М.* // Весці НАН Беларусі. Сер. фіз.-тэх. навук. 2002. № 1. С. 73–76.
20. *Прытков В.А., Садыхов Р.Х., Татур М.М.* // Весці НАН Беларусі. Сер. фіз.-тэх. навук. 2006. № 2. С. 80–86.
21. *Zholtikov R.R., Tatur M.M.* // Int. Scientific Journal of Computing. 2004. Vol. 3, Iss. 1. P. 46–49.
22. *Татур М.М., Островский В.В.* // Вест. фонда фонд. иссл. 2009. № 3. С. 32–41.
23. *Татур М.М., Одинец Д.Н.* // Информатика. 2010. № 3 (27). С. 103–113.
24. *Татур М.М.* // Искусственный интеллект. 2007. № 3. С. 333–343.
25. *Татур М.М., Одинец Д.Н.* // Докл. БГУИР. 2010. № 5 (51). С. 76–81.
26. *Tatur M., Adzinets D., Lukashevich M., Bairak S.* // Advances in intelligent and soft computing. 2010. Vol. 71. P. 529–536.
27. *Татур М.М., Байрак С.А., Одинец Д.Н.* // Технологии безопасности. 2012. № 1 (22). С. 46–47.
28. *Лавникевич Д.А., Татур М.М.* // Докл. БГУИР. 2013. № 6. С. 98–101.
29. *Tatur M.* // J. of Information, Control and Management Systems (Slovakia). 2013. Vol. 11, № 2. P. 155–164.
30. *Белевич А.В., Луцкий В.И., Одинец Д.Н., Татур М.М.* // Радиотехника. 2013. № 2. С. 40–44.
31. *Искра В.В., Искра Н.А., Татур М.М.* // Искусственный интеллект. 2013. № 4 (62). С. 325–332.

СВЕДЕНИЯ ОБ АВТОРЕ



Татур Михаил Михайлович (1959 г.р.), д.т.н., профессор. В 1981 г. закончил МВИЗРУ. В 1988 г. защитил кандидатскую, в 2011 г. – докторскую диссертацию. С 2013 г. – заведующий кафедрой ЭВМ БГУИР. В 2004 г. победил в конкурсе «From Research to Enterprise», проводимом Международной организацией «Central European Initiative». В результате создано предприятие «Интеллектуальные процессоры». Руководил проектами, связанными с построением интеллектуальных и видеоаналитических систем. Автор и соавтор более 120 публикаций, включая 2 монографии, 30 патентов.