

В.В. Спиринцев, Г.А. Бондаренко

ВИКОРИСТАННЯ КЛІТИННИХ АВТОМАТІВ ДЛЯ РОЗПІЗНАВАННЯ СИМВОЛІВ

Анотація. Розглянуто один із пріоритетних напрямків розпізнавання образів, штучного інтелекту та комп'ютерного зору - оптичне розпізнавання символів. Для вирішення цього питання пропонується застосування клітинних автоматів. Досліджено алгоритми (по обробці зображення, поділу зображення тексту на зображення окремих символів, а також алгоритми виділення ознак символів) та наведена їх програмна реалізація на мові програмування Java. Архітектура програми дозволила створювати, зберігати і завантажувати клітинні автомати з мітками та без них, а також послідовності клітинних автоматів.

Ключові слова: розпізнавання, клітинний автомат, клас, мітки.

Постановка проблеми. Задача розпізнавання образів являється однією з найбільш відомих задач комп'ютерного зору, але, не зважаючи на доволі високий рівень дослідів у цій області, на даний момент не існує універсального алгоритму розпізнавання, який був би здатний ефективно вирішити деякі практичні задачі, які мають певну специфіку. Одним з пріоритетних напрямків розпізнавання образів, штучного інтелекту та комп'ютерного зору є оптичне розпізнавання символів (Optical character recognition, OCR) [1,2], оскільки має безліч практичних застосувань: конвертація документів в електронний вигляд, редагування та аналіз інформації, автоматизація систем обліку, архівація документів, автоматичне зчитування чеків, біометричних даних людини, штрих-кодів продуктів, автоматичний переклад на інші мови і т.ін. Тому дослідження в даному напрямку є актуальними та потребують подальшого вивчення.

Аналіз останніх досліджень. Дослідницькою діяльністю в області розпізнавання друкованих і рукописних текстів займаються в усьому світі вже кілька десятиліть поспіль [1,2]. За цей час вдалося створити досить ефективні системи розпізнавання для мов латинської групи, а також для китайської та індійської мов. Точне

розпізнавання символів кирилиці у друкованому тексті зараз можливе тільки, якщо доступні чіткі зображення, такі як друковані документи.

Системи оптичного розпізнавання тексту вимагають калібрування для роботи з конкретним шрифтом. В даний час найбільше поширені так звані «інтелектуальні» системи, з високим ступенем точності розпізнають більшість шрифтів. Серед існуючих програмних рішень оптичного розпізнавання виділяють: Tesseract, Screenwor, АВВУ FineReader, CuneiForm, Ocrad та ін. Також виділяють онлайн-системи розпізнавання, особливість алгоритмів даних систем полягає в тому, що порядок, швидкість і напрямок окремих ділянок ліній введення відомі, а користувач застосовує тільки конкретні форми письма. Ці методи не можуть бути використані в програмному забезпеченні, яке використовує скановані паперові документи, тому проблема розпізнавання рукописного «друкарського» тексту як і раніше залишається відкритою.

Існує великий спектр досліджень та алгоритмів розпізнавання: нейронні мережі, методи гнучкого порівняння графів, приховані марківські моделі, метод головних компонент, активні моделі, використання відстані Гемінга де завдання вибору повністю покладено на дослідника, його навик та переваги [1-3].

Найбільшої популярності для вирішення задач розпізнавання образів набуває застосування штучних нейронних мереж. Суть яких полягає в тому, щоб “навчити” (натренувати) мережу на еталонному зображенні, за яким мережа в майбутньому буде порівнювати вхідні файли шляхом розбиття зображення на площини та порівняння із зображенням, яке збережене у пам’яті [3].

В даних дослідженнях пропонується інший підхід до розв’язання задачі розпізнавання образів - застосування клітинних автоматів (КА). Відмінність КА полягає в тому, що дослідник може виділити унікальні ознаки на етапі навчання системи. Класичні КА в загальному випадку відповідають наступним критеріям [4]: зміна значень всіх клітинок відбуваються одночасно після обчислення нового стану кожної клітинки решітки; решітка однорідна; їх поведінка повністю визначається в термінах локальних залежностей; множина станів клітинки кінцева.

Метою статті є створення моделюючої програми для розпізнавання тексту з використанням КА на базі мови програмування Java.

Основна частина. Для проведення досліджень щодо застосування КА з мітками та без них, а також послідовності КА була розроблена моделююча програма на мові програмування Java.

Кожна сутність в програмі представляється у вигляді класу з набором властивостей і дій, які вона виконувала. КА можна визначити як набір правил. Кожне правило - це сукупність перевірки стану клітини, сусідніх клітин та визначення нового стану клітини. Якщо визначено локальний радіус, на якому сусідні клітини впливають на стан даної, і він дорівнює одиниці, то правило буде містити дев'ять клітин і дія правила - результуюча клітина. Перевірка клітини на задоволення умовами правила розділяється на перевірку стану клітини - його кольору і на перевірку наявності чи відсутності визначених міток. Перевірка кольору клітини може відбуватися декількома способами: клітина повинна мати чітко визначений колір; колір темніший за даний, колір світліший за даний; колір на відрізьку і будь-який колір. Аналогічно перевірка міток задається перевіркою на наявність всіх або однієї із заданих міток, відсутністю всіх або певних міток. Визначення нового стану клітини означає визначення кольору та міток, які будуть визначені для клітини в якості результату виконання правила. Колір клітини може бути залишений колишнім або чітко заданим. Аналогічно, мітки можуть бути залишені колишніми або додані нові. Додатково мітки, знайдені в клітці на етапі перевірки в рамках даного правила можуть бути видалені.

Розглянемо структуру основних частин програми. Моделююча програма необхідна для запуску послідовності клітинних автоматів. Реалізація цього процесу передбачає наявність можливості запуску всієї послідовності в цілому, запуску окремого КА і кроку КА. Крім КА в послідовності можуть бути включені інші елементи: елемент поділу зображення тексту на зображення окремих символів; елемент перевірки умов і переходу за даною умовою на інший елемент послідовності; елемент, який виставляє мітку на задану клітину (початкова умова); елемент, що фіксує результат.

Моделююча програма побудована на посторінковій основі, де сторінки із заданою функціональністю йдуть послідовно одна за іншою. Всього в програмі п'ять сторінок: завантаження зображення; створення і зміна правил клітинних автоматів; комбінування

клітинних автоматів в послідовності; запуску послідовності і розпізнавання тексту щодо даних, отриманих в результаті роботи послідовності клітинних автоматів.

Діаграма класів блоку КА та діаграма класів послідовності КА в системі представлена на рис.1.

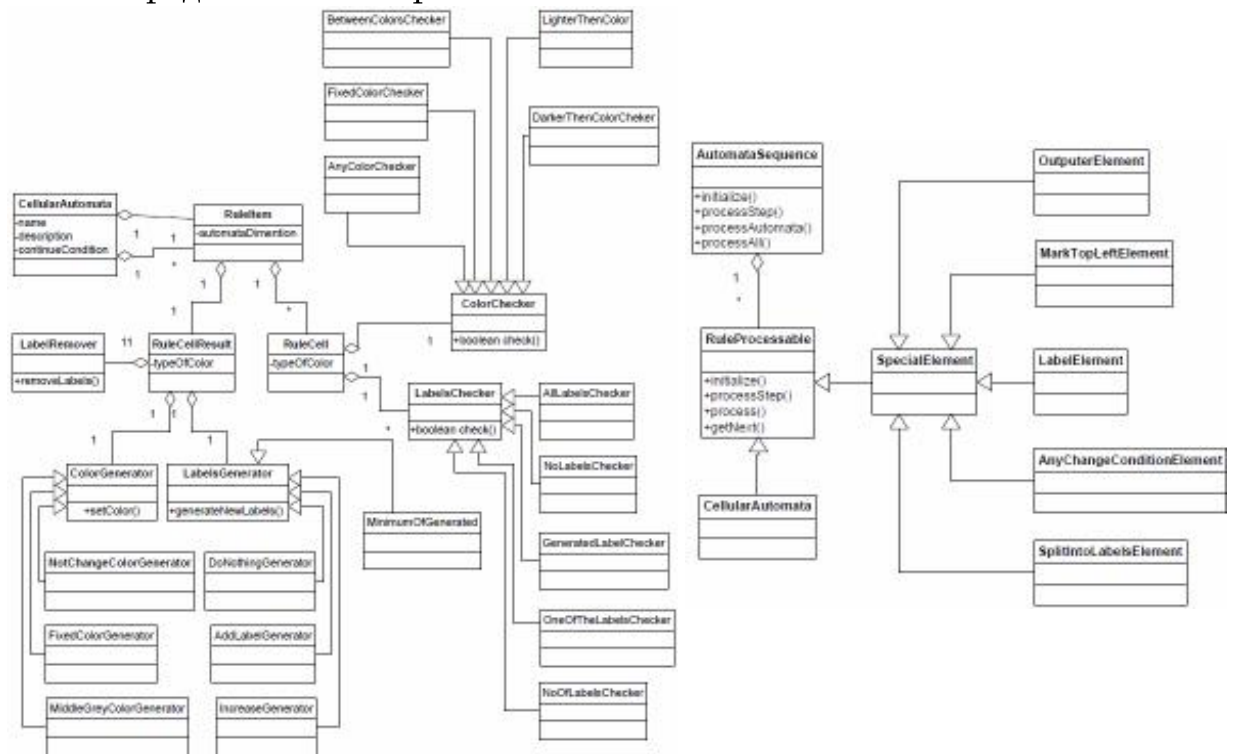


Рисунок 1 – Діаграма класів системи та послідовність КА в системі

Завантаження зображення відбувається на першій сторінці. Зображення є вхідними даними для послідовності КА. На другій сторінці можливе створення КА і їх правил. Список правил відображається у вікні програми, з якого можна вибирати конкретне правило для редагування.

Сторінка комбінування клітинних автоматів дозволяє скласти послідовності запуску КА і визначати додаткову логіку запуску: непослідовні переходи до наступних автоматів за умовою, розбиття зображення, визначення початкової умови для роботи КА.

Після завантаження послідовності з пам'яті програми або з файлу послідовність можна запускати на виконання. При цьому доступні три режими роботи: повний запуск всієї послідовності, запуск автомата на виконання і запуск кроку автомата. На кожному кроці в основному вікні можна переглянути попередні результати роботи.

Навчання системи (рис.2) полягає в запуску алгоритмів виділення ознак символів за спеціально створеним прикладом. Цей

приклад повинен містити, по можливості, всі символи. Найбільш оптимально проводити навчання на зображенні алфавіту. Після обробки послідовністю КА, в кожному зображенні символу алфавіту буде виділено ознаки символів. Користувач програми асоціює набір ознак конкретного символу, і цей зв'язок зберігається. Надалі інформацію про відповідність ознак певним символам можна буде використовувати в процесі розпізнавання текстів, написаних тими ж шрифтами, що й навчальний текст. Після визначення міток навчання запускається кнопкою Запуск. Після цього з'являється спеціальне вікно, в якому користувач вводить символ. Коли всі символи визначені користувачем, дані можна зберегти в файл і потім використовувати в блоці розпізнавання текстів. Після виділення ознак символів на п'ятій сторінці моделюючої програми запускається розпізнавання тексту на основі цих ознак.

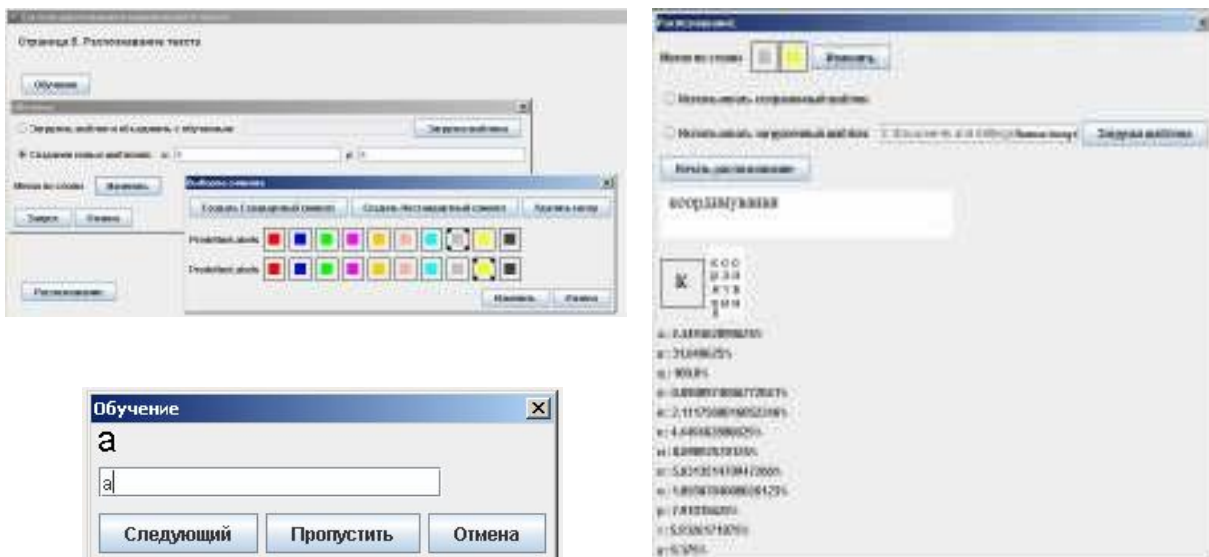


Рисунок 2 – Завдання нових шаблонів та результат розпізнавання

Для цього аналогічно процесу навчання вибираються мітки, що відповідають за ознаки символів, і завантажуються файл шаблону, який містить інформацію навчання на алфавіті. Результатом розпізнавання буде текст, кожен символ якого визначений із заданою точністю на основі статистичного методу (див.рис.2).

В табл. 1 представлена порівняльна характеристика швидкості дії розробленої моделюючої програми на основі КА, нейронної мережі і комерційним продуктом FineReader. Проведений аналіз показав, що FineReader має найменший коефіцієнт похибок (не перевищував 2%),

що визначається використанням комплексу засобів по розпізнаванню, в тому числі, активне використання словників.

Таблица 1

Порівняння часу (с) розпізнавання

Вхідне зображення	КА	Fine Reader	Нейронна Мережа
Зображення слова “координування”	6	1	3
Текст (~800)	92	8	45
Алфавіт, крім літер і, ї, й	15	1	5

Висновки. Розроблена програма дозволяє реалізовувати різні алгоритми для різних шрифтів символів, засновані на КА і КА з мітками, з подальшим запуском і перевіркою працездатності. Зокрема, програма дозволяє виділити ознаки символів, зображень та класифікувати їх. Результати показали стійкість роботи алгоритму для різних розмірів одного і того ж шрифту. У той же час для різних шрифтів алгоритми виділяють дещо відмінні комбінації ознак і їх розташувань, що підтверджує необхідність врахування шрифту при використанні систем OCR на основі даних алгоритмів. Розроблена програма побудована з орієнтацією на розширюваність: врахована можливість додавання нових елементів, необхідних для проведення досліджень. Продуктивність системи, багато в чому, залежить від заданої послідовності клітинних правил і безпосередньо обчислюється з кількості кроків, що виконує КА. Основний час роботи програми займає обхід поля для реалізації роботи КА. Для оптимізації алгоритму обходу слід враховувати принцип локальності правил і реалізовувати принцип паралельного обчислення.

ЛІТЕРАТУРА

1. Травин А. Технологии оптического распознавания текстов [Электронный ресурс]//Газета «Электронный офис».-10.1996.- Режим доступа: <http://travin/msk/ru/arc/OCR.html>.
2. Фролов А. Б. О некоторых подходах к распознаванию оптических образов текстов/А. Б.Фролов, И. Д.Четрафилов// Интеллектуальные системы. 1997. Т. 2. Вып. 1–4. С. 189–200.
3. Нейман Дж. фон. Теория самовоспроизводящихся автоматов. - М.: Мир, 1971. - 382 с.
4. Суясов Д.И. Выделение структурных признаков изображений символов на основе клеточных автоматов с метками [Текст] / Д. И. Суясов// Информационно-управляющие системы. - 2010. - N 4 (47). - С. 39-45.