

Попередня обробка зображень

Котитчук М.Б., Олещук О.В.

Одеський Національний політехнічний університет

65044, пр-т Шевченко 1.

тел. (0482)28-85-37, e-mail: logos104@rambler.ru

Анотація Розглядається спосіб підвищення вірогідності розпізнавання текстових символів, що нанесені на технологічні об'єкти, шляхом попередньої обробки зображення. Пропонується алгоритм, що дозволяє значно скоротити обчислювальні затрати на обробку зображення.

Вступ

Як першоджерела інформації для розпізнавання характерних ознак технологічних об'єктів, використовуються, як правило, відео- та фото зображення [1]. При цьому на вказаних зображеннях можуть бути присутні об'єкти довільного вигляду. тому перед безпосереднім аналізом зображених предметів необхідно виконати ряд операцій, що дозволяють отримати зображення тільки об'єктів

визначеного типу. Існуючі методи обробки зображень [2]-[3] недостатньо враховують специфіку об'єктів спостереження та небажаних об'єктів, що попадають у поле зору.

В рамках даної статті обмежимося розгляданням зображень текстових символів, хоча розглянуті принципи обробки можуть бути застосовані і до деяких інших типів зображень.

Критерії формування характерних ознак

Для вирішення поставленого завдання потрібно визначити, які критерії класифікації можуть бути використані. Через те, що текст звичайно наноситься чорною чи білою фарбою, то можна очікувати, що найбільш темні чи світлі ділянки зображення є символами. На практиці за рахунок поганого освітлення чи наявності різних елементів конструкції та інших об'єктів, що попадають у поле зору, на зображенні часто присутні більш яскраві чи більш темні ділянки, ніж зображення тексту. З тієї ж причини деякі ділянки з текстом не є найбільш контрастними. З іншого боку, в областях, що безпосередньо містять текст, найбільш контрастними ділянками є переходи від зображення символу до фону зображення.

Також необхідно враховувати, що поверхня, на якій зображений текст, не є абсолютно однорідною, як за рахунок нерівномірного фарбування поверхні, так і за рахунок нерівномірного висвітлення.

Виходячи з вищесказаного, виходить, що одним з перших етапів повинне бути виділення фону зображення і вирівнювання яскравості і контрастності по всій області зображення. Для виділення фону необхідно знати колір символів, що

складають напис (чорний чи білий). Через те, що на цьому етапі колір символів невідомий, необхідно зробити потрібні операції для обох кольорів, а вже на більш пізніх етапах виконати визначення кольору символів.

Виділення фону варто робити на невеликих ділянках (чим менше, тим краще, але так, щоб туди не потрапив символ цілком). Спочатку коротко опишемо основні перетворення над матрицями, що використовуються для виділення фону. Умовимося вважати, що код чорного кольору має більше значення, ніж код білого кольору.

1. Поширення максимуму на сусідні точки:

$$D = DMax(S, k_x, k_y) \quad (1)$$
$$D(x, y) = \max(S(u, v))$$

де

$$u = \frac{x - k_x}{2} + 1, x + \frac{k_x}{2} - 1,$$
$$v = \frac{y - k_y}{2} + 1, y + \frac{k_y}{2} - 1.$$

Зміст перетворення $DMax$ полягає в тому, що значення кожного елемента (x,y) матриці D встановлюється в максимальне значення матриці S , знайдене в околиці точки (x,y) . Величина околиці для пошуку максимуму задається параметрами k_x, k_y .

2. Поширення мінімуму на сусідні точки:

$$D = DMin(S, k_x, k_y)$$

$$D(x, y) = \min(S(u, v)), \quad (2)$$

де

$$u = \overline{x - k_x / 2 + 1, x + k_x / 2 - 1},$$

$$v = \overline{y - k_y / 2 + 1, y + k_y / 2 - 1}.$$

Дане перетворення аналогічне перетворенню $DMax$, за винятком того, що виділяє мінімальне значення в заданій області.

Якщо як розмір області задати значення, що перевищують очікувану товщину ліній символів і застосувати операцію $DMax$ над матрицею M_0 із зображенням цифр білого кольору, то на отриманій матриці M_1 області, що раніше містили символи, будуть містити значення, що відповідають кольору фону. Однак при цьому спостерігається деяке збільшення більш темних ділянок зображення за рахунок зменшення світлих ділянок. Це найбільше помітно при яскраво вираженій неоднорідності фону зображення. Щоб позбутися цього небажаного ефекту, потрібно до матриці M_1 застосувати операцію $DMin$ з тими ж параметрами. На отриманій матриці M_2 темні ділянки зображення повернуться до свого вихідного розміру, але на відміну від M_0 в матриці M_2 будуть відсутні світлі ділянки, розміром менше області пошуку мінімумів і максимумів. Якщо тепер з матриці M_2 відняти матрицю M_0 , то буде отримане зображення темних символів на білому фоні.

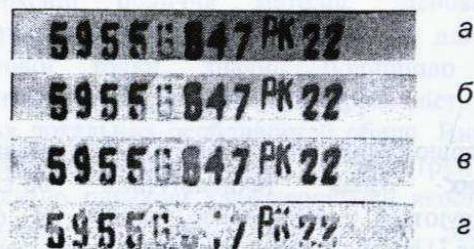
Аналогічно, якщо до зображення з темними символами на світлому фоні послідовно застосувати операції $DMin$ і $DMax$, а потім відняти отриману матрицю з вихідного зображення, то буде отримане зображення темних цифр на білому фоні (мал.1).

Зображення після видалення фону (мал.1г) містить темні ділянки розміром меншим, ніж $k_x * k_y$, де k_x і k_y – параметри в операціях $DMin$ та $DMax$. Вибір параметрів здійснюється виходячи з очікуваної товщини ліній символів на зображенні. Якщо вибрати параметри k_x і k_y занадто великими, то після видалення фону зображення буде містити темні ділянки з фонові області. Чим більше значення параметрів k_x і k_y , тим більше помилкових ділянок фону залишиться на зображенні. При виборі занадто малих значень параметрів k_x і k_y околиці для пошуку

мінімальних і максимальних значень виявляться настільки малими, що лінії символів будуть сприйняті як ділянки фону і будуть вилучені з зображення (мал.2).



Мал. 1. Видалення фону зображення: а – вихідне зображення, б – результат виконання операції $DMin$, в – результат виконання операції $DMax$ над матрицею б, г – різниця зображень а і б.



Мал. 2. Виділення фону з різною величиною околиці пошуку: а – вихідне зображення, б – околиця 64×64 , в – околиця 16×16 , г – околиця 8×8 .

Як правило, вказівка трохи більшого розміру області пошуку не так значно впливає на якість зображення, як область пошуку, менша, ніж товщина лінії символів. Це необхідно враховувати, якщо точно невідома товщина символів чи символи мають різну товщину ліній. У цьому випадку рекомендується вибирати параметри k_x і k_y , виходячи з максимально можливої товщини лінії символів.

Після видалення фону зображення символи, як правило, мають різну яскравість. Це пов'язано як із різною яскравістю символів, так і з різною яскравістю фону на вихідному зображенні (світлі символи на темному фоні будуть яскравіше, ніж світлі символи на світлому фоні). У зв'язку з цим виникає завдання вирівнювання яскравості по всьому зображенню.

Для цього введемо операцію нормалізації області

$$D = NArea(S, k_x, k_y),$$

де

$$D(x, y) = (S(x, y) - M_{Min}(x, y)) / (M_{Max}(x, y) - M_{Min}(x, y)),$$

$$M_{Min} = DMin(S, k_x, k_y),$$

$$M_{Max} = DMax(S, k_x, k_y).$$

(3)

Віднімаючи з матриці S матрицю M_{Min} , тобто мінімальні значення в області, отримаємо, що значення в ній будуть змінюватися від нуля і вище. У знаменнику знаходиться величина, що визначає різницю між максимальним і мінімальним значенням в області обраного розміру. Розділивши на цю величину, одержимо матрицю з діапазоном значень

$$D(x, y) = (S(x, y) - M_{Min}(x, y)) / (M_{Max}(x, y) - M_{Min}(x, y) + \gamma)$$

Недоліком цього способу є залежність результату від співвідношення величини γ і значень матриці S .

$$D(x, y) = \begin{cases} \text{Если } M_{Min}(x, y) \neq M_{Max}(x, y), & \text{то } \frac{S(x, y) - M_{Min}(x, y)}{M_{Max}(x, y) - M_{Min}(x, y)} \\ \text{Если } M_{Min}(x, y) = M_{Max}(x, y), & \text{то } \mu \end{cases}$$

Вибір значення μ варто робити виходячи з особливостей матриці S . Якщо вона є зображенням загального вигляду, то рекомендується встановити μ , рівним 0.5, тобто середині діапазону значень матриці D . Якщо в матриці S попередньо був вилучений фон зображення, то в ній з усіх великих однотонних областей могли залишитися тільки області білого кольору, тобто області із значенням, рівним нулю, тому значення μ варто встановлювати рівним нулю.



Мал. 3. Нормалізація області: а – вихідне зображення, б – нормалізація області розміром 4x4, в – розміром 8x8, г – розміром 16x16, д – розміром 64x16.

В результаті нормалізації області значення кожної точки (x, y) у матриці D буде залежати тільки від значень точок в області розміром $k_x * k_y$, з центром у точці (x, y) . Параметри k_x і k_y вибираються виходячи з очікуваних розмірів символів на зображенні, відстані між ними, розташування символів стосовно інших об'єктів на зображенні (елементів конструкції, різного роду написів та ін.) і виду цих об'єктів. Вибір

від нуля до одиниці. Необхідно передбачити дії при рівності M_{Min} і M_{Max} у деякій точці (x, y) , щоб запобігти ділення на нуль. Можливі дії в цьому випадку:

Додати деяку постійну величину γ у знаменник формули (3).

Інше рішення - у випадку рівності M_{Min} і M_{Max} у деякій точці (x, y) як значення матриці $D(x, y)$ вибрати деяку постійну величину μ .

малих значень k_x і k_y приводить до того, що між символами можуть з'явитися помилкові темні ділянки і з'явиться велика кількість темних ліній у безпосередній близькості до символів, що зробить важчим подальше виділення символів із зображення. При виборі занадто великих значень k_x і k_y , на якість зображення деяких символів можуть впливати яскраво виражені об'єкти, що знаходяться поблизу від символів, у тому числі сусідні символи.

При необхідності після виконання нормалізації області можна перейти до перетворення зображення в монохромне. Це дозволить спростити подальшу обробку і зменшити витрати пам'яті для збереження зображення. Для перетворення зображення в двоколірне необхідно вибрати граничне значення τ , при перевищенні якого точка зображення вважається чорною і їй привласнюється код, рівний одиниці, інакше точка вважається білою з присвоєнням їй коду, рівного нулю.

Виконана послідовність перетворень дозволяє одержати зображення, що містить яскраво виражені об'єкти з заданими геометричними характеристиками. Результат виконання вищеописаних операцій дозволяє легко робити подальшу обробку на основі зв'язності точок, особливо якщо зображення, що спостерігається, містить послідовність недотичних об'єктів. Більш докладно про аналіз монохромних зображень і розпізнавання зображених на них об'єктів див. у [2], [4].

До недоліків розглянутих операцій можна віднести їх відносно велику обчислювальну вартість. Для усунення цього недоліку необхідно оптимізувати алгоритм їхньої реалізації.

Оптимізація операцій

Час виконання операції $DMax(S, k_x, k_y)$ при її виконанні для кожної точки матриці S окремо пропорційний $k_x k_y n_x n_y$,

де n_x, n_y – кількість точок матриці S по осях x та y .

Обчислювальні витрати при виконанні операції поширення максимуму в такому виді істотно залежать від параметрів k_x і k_y , і є дуже значними при великих значеннях k_x і k_y .

Очевидно, що при обчисленні результату операції для двох близько розташованих точок доводиться порівнювати між собою велику кількість тих самих значень, тому доцільно виконувати цю операцію, використовуючи загальні проміжні результати.

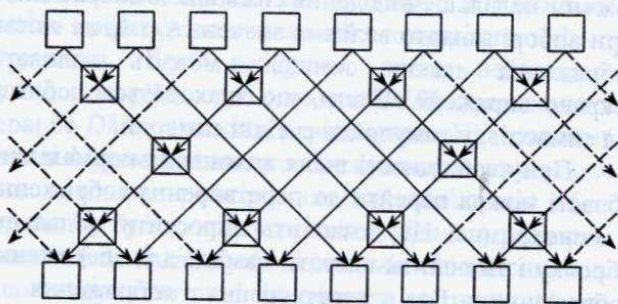
Для простоти розглянемо одномірний вектор V з числом елементів, рівним n ,

де $n = 2^m$, $m = 1, 2, 3, \dots$

і реалізуємо для нього операцію поширення максимуму $DMax(V, k)$.

де $k = 2^p$, $p = 1, 2, 3, \dots$

Запропонована схема реалізації алгоритму показана на мал.4.



Мал.4. Схема реалізації операції поширення максимуму.

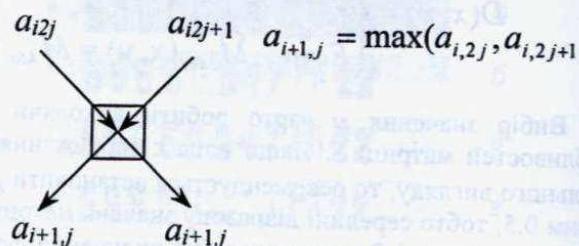
Алгоритм реалізації операції $DMax(V, k)$ умовно можна розділити на два етапи. На першому етапі

здійснюється пошук максимумів серед сусідніх елементів векторів.

На першому кроці попарно порівнюються сусідні елементи вектора V , і максимальне значення з кожної пари записується у вектор V_1 , що має число елементів у два рази менше, чим вектор V . Потім ті ж дії виконуються над вектором V_1 з одержанням вектора V_2 і т.д. доти, поки не буде отриманий вектор V_q ,

де $q = \log_2(k) - 1$.

На другому етапі виконується порівняння елементів векторів різного рівня. Спочатку кожен елемент вектора V_q порівнюється з двома елементами вектора V_{q-1} , що є сусідніми з елементами, з яких отримане значення поточного елемента вектора V_q . Далі отриманий вектор V_{q+1} аналогічно поелементно порівнюється з вектором V_{q-2} і т.д. доти, поки не буде отриманий вектор V_{2q} , що і є шуканим вектором.



Мал.5. Умовне позначення одиничної операції поширення максимуму

Якщо необхідно виконати операцію поширення максимуму, циклічно замикаючи краї матриці, то використовуються зв'язки, зазначені пунктиром. У протилежному випадку пунктирні зв'язки відкидаються.

Швидкість виконання операції поширення максимуму при використанні оптимізованого алгоритму пропорційна $\log(k_x) \log(k_y) n_x n_y$.

Висновки

Запропонований спосіб попередньої обробки зображень символів, нанесених на технологічний об'єкт, дозволяє поліпшити якість зображення, за рахунок чого вірогідність розпізнавання символів зростає до 85%.

Алгоритм оптимізації запропонованих операцій при обробці областей зображень розміром 16×16 дозволяє скоротити час обробки зображень у 16 разів, а при розмірі областей 32×32 – у 40 разів.

Література

1. Копитчук М.Б., Олещук О.В. "Использование метода развертки изображения для распознавания символов" – Праці УНДІРТ, 2001, №3(27). – С. 63 – 67.
2. Дуда Р., Харт П. "Распознавание образов и анализ сцен" – М.: Мир, 1976.
3. Василенко Г.И. Тараторин А.М. "Восстановление изображений" – М.: Радио и связь, 1986.
4. Вапник В.Н., Червонекис А.Я. "Теория распознавания образов" – М.: Наука, 1974.