

ВИСНОВОК

Підвищення продуктивності праці персоналу — найголовніша проблема Укрпошти.

Важливе місце в підвищенні продуктивності праці персоналу Укрпошти посідає впровадження НДР, виконаних НДЦ «Індекс» ОНАЗ ім. О. С. Попова.

Для надання роботі з підвищення продуктивності праці персоналу Укрпошти системного характеру і можливості контролю її перебігу доцільно розробити відповідну цільову програму.

Рецензент: доктор техн. наук, ст. наук. співробітник **М. М. Степанов**, Державний університет телекомунікацій, Київ.

Л. Е. Ящук

ПРОБЛЕМЫ ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ ТРУДА ПЕРСОНАЛА УКРПОШТЫ

Проанализированы причины низкой производительности труда персонала Укрпочты. Обоснованы эффективные пути повышения производительности труда персонала Укрпочты.

Ключевые слова: производительность труда; научно-исследовательские работы (НИР); внедрение результатов НИР.

L. O. Yashchuk

PROBLEMS OF LABOR PRODUCTIVITY INCREASING OF UKRPOSHTA PERSONNEL

The causes for the low labor productivity of Ukrposhta personnel are analyzed. Effective ways of the labor productivity increasing of Ukrposhta personnel are substantiated.

Keywords: labor productivity; research work; implementation of research results.

УДК 004.932.721

О. А. МОКРІНЦЕВ;

В. В. ЖЕБКА,

Державний університет телекомунікацій, Київ

Сучасна методика та алгоритми автоматичного розпізнавання одновимірних штрих-кодів

Висвітлено новітні тенденції та підходи до розробки методики й алгоритмів автоматичного розпізнавання одновимірних штрих-кодів. Досліджено загальну проблематику розпізнавання штрих-кодів. Розглянуто питання їх сегментації та декодування з перевіркою коректності його виконання.

Ключові слова: одновимірний штрих-код; цифрова обробка зображень; сегментація зображень; декодування штрих-кодів.

ВСТУП

Розпізнавання одновимірних штрих-кодів постає як щоденна актуальна потреба в сучасному світі. Вони широко використовуються в поштових відправленнях, логістиці, ритейлі, складській справі та в багатьох інших галузях людської діяльності. Свого часу проблема розпізнавання одновимірних штрих-кодів розв'язувалась переважно за допомогою ручних чи стаціонарних лазерних сканерів, що оперували лише з одновимірним відбитком лінії. Але тепер, у період масового поширення кишенькових комп'ютерів і смартфонів, маємо проблему розпізнавання штрих-кодів, знятих за допомогою цифрових камер, якими зазвичай оснащуються згадані пристрої. При цьому замість розпізнавання ліній довелося визначати й розпізнавати коди у двовимірних зображеннях.

ОСНОВНА ЧАСТИНА**Проблематика розпізнавання штрих-кодів**

Додаткові ускладнення в розпізнаванні виникають через низку проблем з якістю зображення,

зумовлених, зокрема, низькою роздільною здатністю мобільних камер, шумовими явищами, нерівномірним чи незадовільним освітленням, нечітким фокусуванням, колірними спотвореннями та просторовими викривленнями при зйомці.

Загалом процес розпізнавання одновимірних штрих-кодів включає в себе такі кроки:

- 1) попередню обробку зображення;
- 2) локалізацію областей, що містять штрих-код, із визначенням його орієнтації;
- 3) сегментацію, або виділення (сканування), меж ліній штрих-кодів;
- 4) декодування і перевірку коду.

У різних реалізаціях деякі з цих кроків можуть бути скасовані. Проте здебільшого слід виконувати всі зазначені дії, хоча іноді їх послідовність може бути змінена або навіть і довільна.

Перші два кроки реалізують препроцесинг, тобто попередню підготовку зображення до розпізнавання одновимірних штрих-кодів, що має на меті максимізувати шанси на успішне розпізнавання.

Спинимось докладніше на алгоритмах і методах, що застосовуються при виконанні двох останніх кроків — власне процесингу, або розпізнавання штрих-кодів.

Сегментація штрих-кодів, або виділення меж їхніх ліній

На цьому етапі вважається, що ми маємо послідовність точок зображення у відтинках сірого (рівні яскравості), при цьому ці точки утворюють відрізок, який потенційно перетинає всі лінії штрих-коду. Потрібно виявити у вхідних даних послідовність чорних і білих штрихів різної товщини. Лінії, що утворюють штрих-код, залежно від типу кодування можуть або просто поділятися на тонкі і товсті, або бути одинарної, подвійної, потрійної (чи навіть ще більшої кратності) товщини [1]. Здебільшого задача сегментації зводиться до відшукування деякого порогового значення, яке, у свою чергу, може бути або статично глобальне, або адаптивно-локально змінне, але таке, що дозволяє визначити межі білих і чорних штрихів.

Варто зазначити, що не всі методи сегментації (бінаризації) прийнятні з погляду застосування в термінах поставленої задачі. Наприклад, метод Яновіца–Брукштейна [9], який дає цілком задовільні результати в розпізнаванні сканованих картографічних документів, не може бути використаний. Адже зображення в процесі їх обробки фільтруються для отримання контурних ліній завтовшки в піксель, а далі ще й додатково усереднюються.

До найбільш поширених і перспективних методів, що можуть бути використані на зазначеному етапі, належать сім розглянутих далі методів.

1. Сканування локальних мінімумів і максимумів. Цей метод найпростіший у реалізації, хоча й має серйозні недоліки. Так, якщо зображення забруднено дрібним шумом («пил»), то через це виникає багато фальшивих піків і проміжків. Щоб зменшити вплив цього явища, зазвичай застосовують попередню фільтрацію зображення (наприклад, методом розмиття).

2. Метод збалансованого відсікання гістограми. При обробці зображень алгоритм збалансованого порогового відсікання гістограми [2] використовується для бінаризації півтонових об'ємів. У цьому алгоритмі передбачається, що зображення містить пікселі двох класів: ті, що належать фону, і переднього плану. Як і метод Оцу та метод ітеративного вибору порога, він спирається на ітеративне відшукування порогового значення, яке поділяє пікселі на зазначені класи. У цьому методі зважуються дві рівні частини гістограми. Якщо одна частина переважає, то з неї вилучається крайній стовпчик. Ітеративна процедура закінчується, коли в гістограмі залишається лише один

стовпчик. При цьому відповідне йому значення інтенсивності вибирається як порогове значення.

3. Метод Ейквіла (рис. 1). Цей метод — один із найефективніших щодо обробки чітких і контрастних зображень.

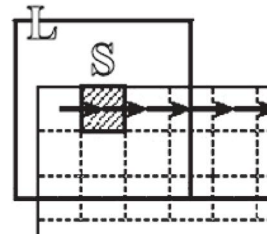


Рис. 1. Ілюстрація методу Ейквіла

Зображення обробляється за допомогою двох концентричних вікон: маленького S і великого L . Зазвичай форма вікон береться квадратна. Обидва вікна послідовно зліва направо та згори вниз накладаються на зображення з кроком, що дорівнює стороні вікна S . Для вікна L розраховується поріг B так, щоб поділити пікселі на два кластери. Якщо різниця значень математичного сподівання рівнів яскравості в двох кластерах перевищує деякий заданий користувачем рівень $(\mu_1 - \mu_2) \geq l$, то всі пікселі всередині вікна S визначаються як чорні або білі відповідно до порогового значення T . У разі невиконання цієї умови яскравість пікселів з вікна S замінюється деяким близьким проміжним (інтерпольованим) значенням.

4. Метод Бернсена. Цей метод [3] використовує заданий попередньо поріг E контрастності. Якщо локальний контраст $(\max - \min)$ перевищує або дорівнює заданому значенню, поріг встановлюється як локальне середнє у відтинках сірого (із мінімального і максимального значень сірого в локальному вікні навколо поточного пікселя). Далі, порівнюючи з установленим порогом поточний піксель, визначаємо його колір — чорний або білий.

У тому разі, коли локальний контраст нижчий від попередньо заданого E , уся локальна область вважається монотонною і вихідний колір пікселя встановлюється як значення, задане для «сумнівних» областей.

Метод має низку недоліків. Після обробки монотонних областей яскравості формуються сильні паразитні шуми, а в деяких випадках це призводить до появи небажаних чорних плям.

5. Бінаризація за методом Оцу. Цей метод [4] дозволяє ефективно обчислювати порогове значення для півтонових зображень. За допомогою цього алгоритму у його класичному вигляді можна розрізнити пікселі двох класів (чорні або білі) побудовою серії гістограм та мінімізацією внутрішньокласової дисперсії стосовно порога. Ідея полягає у відшуванні такого порога, що в межах класу мінімізує дисперсію, котра визначається як зважена сума дисперсій двох класів:

$$\sigma_{\omega}^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t),$$

де $\omega_i(t)$ — імовірність одного з двох класів, відокремлених пороговим значенням t , а $\sigma_i(t)$ — дисперсія цього класу. Оцу довів, що мінімізація дисперсії всередині класу еквівалентна максимізації міжкласової дисперсії яка подається у вигляді

$$\sigma_b^2(t) = \sigma^2 - \sigma_{\omega}^2(t) = \omega_1(t)\omega_2(t)[\mu_1(t) - \mu_2(t)]^2$$

і визначається через імовірність ω_i і середнє арифметичне значення μ_i , яке можна ефективно оновлювати в ітераційний спосіб. Загалом, метод працює досить швидко й ефективно, але в разі випадку нерівномірно освітлених зображень глобальне порогове значення може дати хибні результати.

6. Бінаризація за методом Ніблека. Цей метод [5] базується на обчисленні локального середнього значення та локального середньоквадратичного відхилення. Порогове значення знаходимо за формулою

$$B(x, y) = \mu(x, y) + k \times s(x, y),$$

де $\mu(x, y)$ — середнє значення в деякій локальній області, а $s(x, y)$ — відповідне стандартне відхилення. Розмір такої області має бути достатньо малий, щоб коректно відбивати особливості локальних деталей, а водночас і достатньо великий, щоб уникнути шумових явищ. Параметр k визначає, яку частину меж об'єкта розглядаємо як сам об'єкт.

Жанг і Тан [6] запропонували дещо вдосконалений варіант цієї формули:

$$B(x, y) = \mu(x, y) \times \left[1 + k \times \left(1 - \frac{s(x, y)}{R} \right) \right],$$

де k і R — деякі емпіричні константи. У такому вигляді алгоритм менш чутливий до шумових явищ.

7. Ітеративний метод Рідлера-Калварда (isodata). Цей метод [7] рекомендовано до застосування у випадках зашумлених і слабо контрастних зображень. Спочатку ми в деякий спосіб, наприклад із мінімуму і максимуму гістограми, беремо деякий початковий поріг t_0 , який дорівнює просто середньому рівню освітленості зображення. Далі за площею фону та переднього плану ми обчислюємо два середні значення рівня сірого для кожної з цих двох областей. А нове порогове значення t_{k+1} беремо як середнє арифметичне обчислених середніх відтінків сірого:

$$t_{k+1} = \frac{\mu_1(t_k) + \mu_2(t_k)}{2},$$

де μ_1 — усереднений відтінок сірого фону, а μ_2 — усереднений відтінок сірого фігури (переднього плану). Згідно з новим порогом проводимо нову бінаризацію зображення. Продовжуємо цей процес ітеративно доти, доки дельта між t_k і t_{k+1} не зменшиться до фактичної тотожності цих значень. Автори методу відзначають, що зазвичай вистачає виконання трьох-чотирьох ітерацій.

Декодування і перевірка коду

На цьому етапі ми декодуємо знайдену послідовність штрихів і проміжків у цифровий код. Перевірка коректності декодування включає в себе:

- розпізнавання «тихих» зон, стартової та фінішної контрольних комбінацій штрихів і проміжків;
- успішне розпізнавання всіх сканованих комбінацій цифр (чи символів);
- перевірку контрольної суми.

Здебільшого як контрольну суму використовують одну чи дві цифри. Скажімо, для наведеного на рис. 2 штрихового коду EAN 13 [8] контрольна цифра обчислюється за алгоритмом, який являє собою один із різновидів алгоритму Луна, що включає в себе такі кроки.



Рис. 2. Приклад штрих-коду EAN-13

1. Цифрам у штрих-кодi ставлять у відповідність номери 1–12 згідно з їх позицією, виключаючи останню контрольну цифру.

2. Використовують додавання цифр, розташованих на парних місцях, і знайдену суму множать на 3. Наприклад: $9 + 1 + 3 + 1 + 3 + 5 = 22$; $22 \cdot 3 = 66$.

3. Додають цифри, розташовані на непарних місцях.

Наприклад: $5 + 0 + 2 + 4 + 2 + 4 = 17$.

4. Додають результати, здобуті в пунктах 2 і 3, дістаючи дво- чи трицифрове число.

Наприклад: $66 + 17 = 83$.

5. В остаточній сумі залишають лише те число, яке розташовується на останньому місці.

Наприклад: 3.

6. Віднімають від числа 10 число, знайдене в попередньому пункті. Здобута різниця і є контрольною цифрою. Вона має збігатися з цифрою, зазначеною в штрих-кодi.

Наприклад: $10 - 3 = 7$.

Можна застосовувати й інші методи розрахунку контрольних сум (цифр), наприклад за допомогою циклічного надлишкового коду.

Теоретично для надкритичних ситуацій можуть бути використані методи корекції помилок, такі як коди Ріда-Соломона. Але на практиці штрих-коди й так містять настільки обмежену кількість інформації, що немає сенсу переважувати її надлишковими даними, ускладнюючи відновлення. Тому зазвичай обмежуються контрольною сумою чи навіть однією цифрою.

ВИСНОВКИ

Для розпізнавання одновимірних штрих-кодів у попередньо оброблених зображеннях, які щонайменше перетворені з кольорових у відтінки сірого, зазвичай виконують:

◆ сегментацію — виділення (сканування) меж ліній штрих-кодів;

◆ декодування і перевірку коду.

Мета сегментації — визначити межі білих і чорних штрихів для подальшого декодування. Найчастіше для цього застосовують сканування локальних максимумів і мінімумів; метод збалансованого відсікання гістограми; методи бінаризації Ейквіла, Бернсена, Оцу, Ніблека, а також ітеративний метод Рідлера–Калварда (isodata).

Процес декодування має бути верифікований перевіркою контрольного коду (цифри), котрий здебільшого обчислюється за допомогою деякої модифікації алгоритму Луна.

Список використаної літератури

1. **Pavlidis, T.** *Fundamentals of bar code information theory* / T. Pavlidis, J. Swartz, and Y. P. Wang // *Computer*.— Apr. 1990.— Vol. 23, no. 4.— P. 74–86.

2. **Anjos, A.** *Bi-Level Image Thresholding — A Fast Method* / A. Anjos, H. Shahbazkia // *BioSignals*.— 2008.— Vol. 2.— P. 70–76.

3. **Bernsen, J.** *Dynamic thresholding of grey-level images* / J. Bernsen // *Proc. 8th Int. Conf. on Pattern Recognition, Paris, 1986*.— P. 1251–1255.

4. **Otsu, N.** *A Threshold Selection Method from Gray-Level Histograms* / N. Otsu // *Automatica*.— 1975.— Vol. 11.— P. 285–296.

5. **Niblack.** *An Introduction to Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1986.— P. 115–116.

6. **Zhang, Z.** *Restoration of images scanned from thick bound documents* / Z. Zhang and C. L. Tan // *Proc. Int. conf. Image Processing*.— 2001.— Vol. 1.— P. 1074–1077.

7. **Ridler, T. W.** *Picture Thresholding Using an Iterative Selection Method* / T. W. Ridler and S. Calvard // *IEEE Trans. on Systems, Man and Cybernetics*.— 1978.— Vol. SMC-8, no. 8.— P. 630–632.

8. <https://www.wikipedia.org/>

9. **Leedham et al.** *Comparison of some thresholding algorithms for text/background segmentation in difficult document images* // *IEEE*.— 2003.

Рецензент: доктор техн. наук, професор А. І. Семенко, Державний університет телекомунікацій, Київ.

А. А. Мокринцев, В. В. Жебка

СОВРЕМЕННАЯ МЕТОДИКА И АЛГОРИТМЫ АВТОМАТИЧЕСКОГО РАСПОЗНАВАНИЯ ОДНОМЕРНЫХ ШТРИХ-КОДОВ

Освещены новейшие тенденции и подходы к разработке методики и алгоритмов автоматического распознавания одномерных штрих-кодов. Исследована общая проблематика распознавания штрих-кодов. Рассмотрены вопросы их сегментации и декодирования с проверкой корректности его выполнения.

Ключевые слова: одномерный штрих-код; цифровая обработка изображений; сегментация изображений; декодирование штрих-кодов.

O. A. Mokrintsev, V. V. Zhebka

MODERN METHODS AND ALGORITHMS OF COMPUTER LINEAR BARCODE RECOGNITION

The work covers latest tendencies and approaches in linear barcodes reading methods development. Studied problems barcode recognition. Considered segmentation or isolation boundaries lines barcodes. Investigated decoding and verification code.

Keywords: one-dimensional barcode; digital image processing; image segmentation; barcode decoding.

УДК 621.391

А. О. МАКАРЕНКО, канд. техн. наук, доцент;

Г. О. ГРИНКЕВИЧ, канд. техн. наук, доцент;

Н. В. КОРШУН, канд. техн. наук, доцент;

В. М. КУКЛОВ, аспірант;

А. І. ПІДРУЧНИЙ, аспірант;

Г. В. ТКАЧ, студентка,

Державний університет телекомунікацій, Київ

Аналіз програмного забезпечення, необхідного для контролю пакетів даних у програмно-конфігурованих мережах

Розглянуто питання щодо вимірювання та моніторингу Software-Defined-Networking, покладені в основу алгоритмів виявлення несправностей, і подано огляд сучасних програмних засобів, які мають на меті забезпечити безпеку й контроль трафіку в програмно-конфігурованих мережах.

Ключові слова: інформаційні мережі; програмно-конфігуровані мережі; інформаційна безпека; SDN; Linux; Ethernet; OpenFlow.

ВСТУП

Постановка задачі. Техніка захисту може відновити OpenFlow мережі в межах 50 мс. Для цього жодних дій від мережного контролера не знадобиться, бо перемикач має змогу безпосередньо реагувати на несправності. У техніці реставрації контролер має взаємодіяти з мережними пристроями, що потребує більше часу і робить метод менш зручним для великомасштабних мереж із багатьма підвузлами.

© А. О. Макаренко, Г. О. Гринкевич, Н. В. Коршун, В. М. Куклов, А. І. Підручний, Г. В. Ткач, 2017