

В роботі розглянуті типові алгоритми для виділення границь об'єктів на зображенні та досліджені особливості застосування операторів згортки в алгоритмі Кенні. В результаті експериментально встановлено, що застосування оператора Собеля є оптимальним. В якості альтернативних операторів згортки були також розглянуті оператор Робертса та Превітта. В результаті наведено рекомендації щодо реалізації алгоритму Кенні

Ключові слова: алгоритм Кенні, оператор Собеля, оператор Робертса, оператор Превітта, виділення границь, градієнт

В работе рассмотрены типичные алгоритмы для выделения границ объектов на изображении и исследованы особенности применения операторов свертки в алгоритме Кенни. Экспериментально установлено, что применение оператора Собеля является оптимальным. В качестве альтернативных операторов свертки были также рассмотрены операторы Робертса и Превитта. В результатах приведены рекомендации по реализации алгоритма Кенни

Ключевые слова: алгоритм Кенни, оператор Собеля, оператор Робертса, оператор Превитта, выделение границ, градиент

УДК 007.51:611.81

DOI: 10.15587/1729-4061.2015.56548

ДОСЛІДЖЕННЯ ЗАСТОСУВАННЯ ОПЕРАТОРІВ ЗГОРТКИ В ЗАДАЧАХ ВИДІЛЕННЯ ГРАНИЦЬ НА ЗОБРАЖЕННІ

С. О. Петров

Кандидат технічних наук, доцент*

E-mail: sergpet@gmail.com

І. О. Марченко

Аспірант*

E-mail: mar4enko.igor@gmail.com

Б. О. Дібров

Аспірант*

E-mail: dibrov.bor@gmail.com

*Кафедра комп'ютерних наук

Сумський державний університет

вул. Римського-Корсакова, 2, м. Суми, Україна, 40007

1. Вступ

Виділення границь є однією з важливих задач комп'ютерного зору [1]. Зазвичай це є проміжною ланкою іншого алгоритму автоматичної обробки зображень в задачах розпізнавання. Складність вирішення цієї задачі обумовлюється чутливістю методів до шумів, дисперсії яскравості та до перетину об'єктів [2]. Враховуючи нестаціонарність умов отримання вхідних зображень та високу чутливість алгоритмів їх обробки до шумів у вхідних даних виникає необхідність побудови детермінованого процесу застосування того чи іншого алгоритму.

Виділення границь знаходить широке застосування для вирішення наступних практичних задач: розпізнаванні номерів автомобілів [3], OCR, медицині [4], картографії [5], кластеризації, знаходження тексту, фігур або складних об'єктів, наприклад, людей на зображенні [6, 7]. Слід зазначити, що при практичному застосуванні алгоритму Кенні дослідники зосереджувались на аналізу кожного виду вхідних даних окремо, не порівнюючи отримані результати між різними типами зображень.

2. Аналіз літературних даних та постановка задачі

Суттєві результати в розв'язку даної задачі досягнуті сучасними дослідниками в роботах [8–11], які пов'язані з лікуванням ракових захворювань, діагностування тромбозу вен, результатами МР-ангіографії та для створення автопілотів для автомобілів [12].

Мета пошуку границь у знаходженні границь (кривих) зміни яскравості або кольору [7]. В подальшому з границь будуються контури об'єктів, а з них – сцени. Вхідними даними для алгоритму є зображення $I_{x,y}$ кожен елемент якого містить три компоненти кольору – червону, зелену і синю, що позначаються $red(I_{x,y})$, $green(I_{x,y})$, $blue(I_{x,y})$. Результатом роботи алгоритму є матриця $R_{x,y}$, кожен елемент якої дорівнює 1, якщо піксель з координатами (x, y) є границею, або 0 в іншому випадку [1].

Розв'язок цієї задачі повинен відповідати вимогам [13, 14] з яких окремо виділимо наступні:

- алгоритм має знайти якомога більше границь присутніх на зображенні;
- знайдені границі мають бути максимально близькі до реальних границь;
- одна реальна границя не повинна створювати декілька паралельних границь на виході;
- шуми не повинні створювати неіснуючих границь.

Для перевірки результату роботи алгоритму необхідно знайти реальні границі об'єктів на зображенні та побудувати еталонну матрицю $E_{x,y}$, після чого обрахувати міру схожості матриць ϕ за формулою

$$\phi = \frac{1}{|E_{x,y}|} \cdot \sum_{i,j} \begin{cases} 0, & \text{якщо } E_{x,y} = R_{x,y}, \\ 1, & \text{якщо } E_{x,y} \neq R_{x,y}, \end{cases} \quad (1)$$

де $|E_{x,y}|$ – кількість елементів в матриці E .

Виходячи з наведеного огляду можна зробити висновки, що при застосуванні алгоритму дослідники проводять підлаштування вхідних зображень для досягнення високої результативності алгоритму для обробки певного класу зображень, але при цьому не враховують можливості підвищення ефективності роботи алгоритму Кенні шляхом вибору оптимального ядра згортки.

3. Мета і завдання дослідження

Метою роботи є дослідження впливу оператора згортки в алгоритмі Кенні в контексті розв'язку задачі виділення границь об'єкту на зображенні та аналіз його фактичних результатів в загальному випадку.

Для досягнення поставленої мети дослідження було сформульовано наступні завдання:

- провести математичний опис оператора згортки в алгоритмі Кенні;
- визначити критерій оптимальності роботи оператора згортки як міру схожості результуючих матриць та розробити алгоритм його обчислення;
- провести експериментальне дослідження результативності роботи алгоритму на еталонній базі вхідних зображень [15] визначивши оптимальний оператор згортки.

4. Методи розв'язання задачі дослідження

Домінуючими методами пошуку границь є детектори Робертса [16], Собеля [4], Мар-Хільдрета [4], Кенні [7]. Завдяки тому, що зображення представляє собою матрицю пікселів, дані методи базуються за пошуку залежностей між пікселем і його сусідами.

Виконаємо модифікацію алгоритму Кенні. Основні етапи базового алгоритму:

1. Перетворення кольорового зображення у відтінки сірого

Кольорове зображення конвертується у відтінки сірого за формулою

$$I_{x,y} = 0.212 \cdot \text{red}(I_{x,y}) + 0.715 \times \\ \times \text{green}(I_{x,y}) + 0.072 \cdot \text{blue}(I_{x,y}).$$

Константи перед відповідними компонентами кольору $\text{red}(I_{x,y})$, $\text{green}(I_{x,y})$ та $\text{blue}(I_{x,y})$ отримані емпірично, з урахуванням фізіологічних особливостей сприйняття кольору людиною [17].

2. Видалення шумів

Алгоритм чутливий до шумів у зображенні тому, для видалення шуму використаємо фільтр Гауса [13]

$$S_{x,y} = I_{x,y} * G_{x,y,\sigma},$$

де $S_{x,y}$ – вихідний піксель; $I_{x,y}$ – вхідний піксель; x, y – координати пікселя; $G_{x,y,\sigma}$ – фільтр Гауса, який задається формулою

$$G_{x,y,\sigma} = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}},$$

σ – коефіцієнт розсіювання; ρ – розмірність ядра; * – операція згортки.

Розглянемо детально роботу алгоритму. Для прикладу візьмемо коефіцієнт $\sigma=1.2$ і ядро $\rho=5$. Отримаємо фільтр

$$G = \begin{pmatrix} 0.00825 & 0.0233 & 0.033 & 0.0233 & 0.00825 \\ 0.0233 & 0.0662 & 0.0937 & 0.0662 & 0.0233 \\ 0.033 & 0.0937 & 0.132 & 0.0937 & 0.033 \\ 0.0233 & 0.0662 & 0.0937 & 0.0662 & 0.0233 \\ 0.00825 & 0.0233 & 0.033 & 0.0233 & 0.00825 \end{pmatrix}.$$

Розмір ядра впливає на відстань (у пікселях), на яку буде проводитись розсіювання. Різноманітні матриці, сгенеровані даним методом, використовуються для надання зображенню графічних ефектів [18].

3. Пошук градієнту зміни яскравості

Градієнт

$$\nabla S_{x,y} = \left(\frac{\partial S_{x,y}}{\partial x}, \frac{\partial S_{x,y}}{\partial y} \right)$$

показує напрям і інтенсивність збільшення яскравості. Для кожного пікселя обчислюється величина градієнту $M_{x,y}$ і напрям $\theta_{x,y}$. Напрямок градієнту округлюється до одного з восьми кутів $\frac{\pi}{4} \cdot i$, для $i \in [0,8)$.

$$M_{x,y} = \sqrt{\left(\frac{\partial S_{x,y}}{\partial x} \right)^2 + \left(\frac{\partial S_{x,y}}{\partial y} \right)^2};$$

$$\theta_{x,y} = \tan^{-1} \left(\frac{\frac{\partial S_{x,y}}{\partial y}}{\frac{\partial S_{x,y}}{\partial x}} \right).$$

Для обчислення градієнту на зображенні використовуються оператори згортання. Обчислимо градієнти трьома різними операторами.

а. Оператор Собеля

$$\frac{\partial S_{x,y}}{\partial x} = S_{x,y} * \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix},$$

$$\frac{\partial S_{x,y}}{\partial y} = S_{x,y} * \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}.$$

Типовий оператор для детектору Кенні. Виконує згортку, беручи яскравість пікселів у напрямі градієнту двічі, тобто приділяючи їм більшої ваги. Результатом застосування оператора Собеля є неточне наближення градієнту зображення.

б. Оператор Превітта

$$\frac{\partial S_{x,y}}{\partial x} = S_{x,y} * \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix},$$

$$\frac{\partial S_{x,y}}{\partial y} = S_{x,y} * \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}.$$

Оператор Превітта однаково враховує пікселі по напрямку градієнту і по діагоналям (горизонтальному, або вертикальному). На практиці оператор Превітта використовується для пошуку величини та орієнтації границі.

в. Оператор Робертса

$$\frac{\partial S_{x,y}}{\partial x} = S_{x,y} - S_{x+1,y+1};$$

$$\frac{\partial S_{x,y}}{\partial x} = S_{x+1,y} - S_{x,y+1}.$$

На відміну від попередніх операторів, оператор Робертса обраховує градієнт у двох діагональних напрямках. Результатом застосування є градієнт від двох верхніх пікселів до двох нижніх. До переваг данного оператора можна віднести швидкість його роботи. Головним недоліком є чутливість до шумів.

Застосуємо оператор Собеля на матриці яскравості

$$I = \begin{pmatrix} 123 & 43 & 67 & 107 & 150 \\ 228 & 80 & 158 & 172 & 167 \\ 43 & 243 & 31 & 154 & 97 \\ 23 & 193 & 237 & 20 & 8 \\ 243 & 100 & 178 & 189 & 120 \end{pmatrix}.$$

Використовуючи фільтр Гауса отримаємо матрицю $S_{x,y}$

$$S = \begin{pmatrix} 60.15375 & 75.0805 & 84.0266 & 85.44045 & 69.79755 \\ 86.78795 & 115.3525 & 126.20345 & 119.53315 & 91.33915 \\ 97.69405 & 138.308 & 149.0093 & 130.53675 & 91.70485 \\ 94.0071 & 136.4508 & 145.89985 & 121.5148 & 79.7783 \\ 75.11045 & 105.6605 & 112.8296 & 93.99325 & 61.37255 \end{pmatrix}.$$

Після виконання оператора Собеля отримаємо матрицю

$$\frac{\partial S_{x,y}}{\partial x} = \begin{pmatrix} 207.095 & 265.514 & 87.161 & 24.901 & -63.322 \\ 331.424 & 444.094 & 154.019 & 10.95 & -141.262 \\ 376.183 & 528.419 & 193.939 & -26.298 & -215.595 \\ 360.819 & 516.87 & 192.82 & -49.311 & -241.005 \\ 244.228 & 347.772 & 127.331 & -38.271 & -169.036 \end{pmatrix};$$

$$\frac{\partial S_{x,y}}{\partial y} = \begin{pmatrix} 86.788 & 288.928 & 443.696 & 487.293 & 456.609 \\ 37.54 & 138.308 & 228.978 & 238.289 & 177.083 \\ 7.219 & 35.537 & 69.112 & 62.473 & 12.099 \\ -22.584 & -77.815 & -124.058 & -141.55 & -139.599 \\ -94.007 & -324.465 & -512.809 & -549.765 & -468.708 \end{pmatrix};$$

$$M = \begin{pmatrix} 224.545 & 392.399 & 452.176 & 487.928 & 460.979 \\ 333.543 & 465.132 & 275.958 & 238.541 & 226.524 \\ 376.252 & 529.613 & 205.885 & 67.782 & 215.934 \\ 361.525 & 522.695 & 229.281 & 149.893 & 278.516 \\ 261.696 & 475.629 & 528.38 & 551.096 & 498.257 \end{pmatrix};$$

$$\theta = \begin{pmatrix} \frac{\pi}{2} & \frac{\pi}{4} & 0 & 0 & 0 \\ \frac{\pi}{2} & \frac{\pi}{2} & \frac{\pi}{4} & 0 & \frac{7\pi}{4} \\ \frac{\pi}{2} & \frac{\pi}{2} & \frac{\pi}{2} & 0 & \frac{3\pi}{2} \\ \frac{\pi}{2} & \frac{\pi}{2} & \frac{3\pi}{4} & \pi & \frac{5\pi}{4} \\ \frac{3\pi}{4} & \frac{3\pi}{4} & \pi & \pi & \pi \end{pmatrix}.$$

4. Пригнічення не-максимумів

Для кожного пікселя, якщо величина його градієнта максимальна у напрямку градієнту, то він залишається без змін, інакше його величина приймається рівною нулю.

Наприклад, якщо

$$\theta_{x,y} = \frac{\pi}{2} \text{ і } M_{x,y} < M_{x,y-1},$$

то $M_{x,y}$ анулюється.

5. Відсіювання по граничним значенням

За двома параметрами *Low* та *High* анулюються градієнти, величина яких не входить до інтервалу [*Low*; *High*]. На завершення у результуючій матриці $R_{x,y}$ границями позначаються такі пікселі, у яких величина градієнту не нульова

$$R_{x,y} = \begin{cases} 0, & \text{якщо } M_{x,y} = 0, \\ 1, & \text{якщо } M_{x,y} \neq 0. \end{cases}$$

Результатом роботи алгоритму є бінарна матриця, в якій пікселям, що належать границі об'єкту відповідає значення 1, а всім іншим – 0.

5. Застосування розробленого алгоритму

Для наглядної демонстрації застосування розробленої моделі розглянемо наступні 3 зображення (рис. 1, а-в), як загальний випадок вхідного зображення, алгоритми для обробки якого описані в роботі.



Рис. 1. Вхідні зображення для проведення експерименту: а – герб СумДУ; б – Lena; в – кольорове коло

Зображення вибрані таким чином, щоб забезпечити представлення таких груп зображень як векторні логотипи (рис. 1, а), зображення з великою кількістю деталей (рис. 1, б), та чіткі зображення різноманітні за кольором (рис. 1, в). Результати роботи методу Кенні показані на рис. 2.

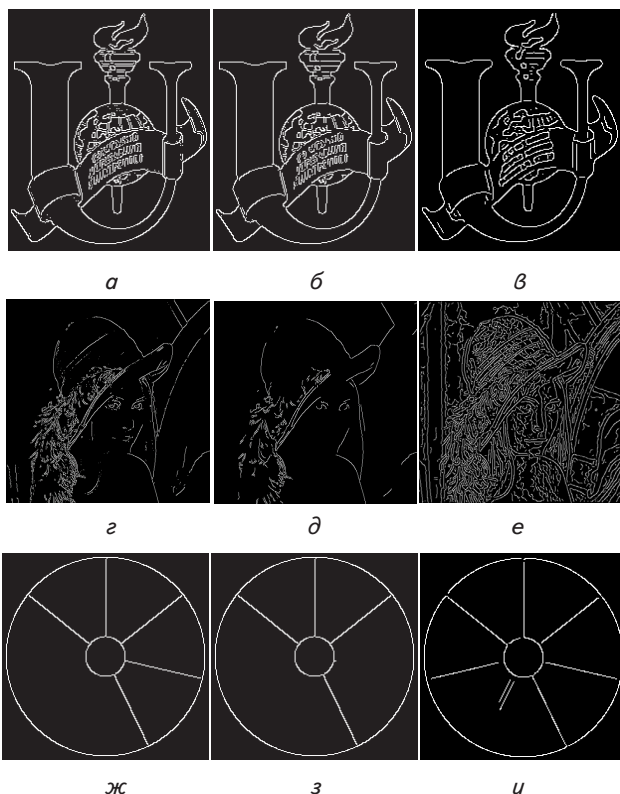


Рис. 2. Результати роботи методів з модифікаціями:

- а – оператор Собеля Low=320, High=370;
- б – оператор Превітта Low=320, High=370;
- в – оператор Робертса Low=320, High=370;
- г – оператор Собеля Low=200, High=250;
- д – оператор Превітта Low=200, High=250;
- е – оператор Робертса Low=200, High=250;
- ж – оператор Собеля Low=250, High=300;
- з – оператор Превітта Low=250, High=300;
- и – оператор Робертса Low=250, High=300

У векторному зображенні оператор Собеля (рис. 2, а) виділив важливі границі і не пропустив деталей, оператор Превітта (рис. 2, б) дав схожий результат, але з менш чіткими границями, оператор Робертса (рис. 2, в) не виділив прямі лінії і заокруглив прямі кути. У зображенні з великою кількістю деталей оператор Собеля (рис. 2, г) знайшов основні контури обличчя та капелюха, але пропустив менш яскраві деталі інтер'єру, оператор Превітта (рис. 2, д) не знайшов основні контури обличчя, а оператор Робертса (рис. 2, е) виділив багато незначних контурів. Для зображення з різноманітними кольорами оператор Собеля (рис. 2, ж) і Превітта (рис. 2, з) не знайшли деякі границі, а оператор Робертса (рис. 2, и) знайшов усі границі.

Обчислимо міру схожості для операторів Собеля та Робертса. Для прикладу візьмемо нижню границю $Low=High-50$, тоді узагальнені значення параметру $High$ та відповідна йому міра схожості ϕ для гербу СумДУ (рис. 1, а) представлені на рис. 3, 4.

З рис. 3, 4 видно, що оператор Собеля при однакових границях має більше значення міри схожості ніж оператор Робертса, а також швидше його набирає.

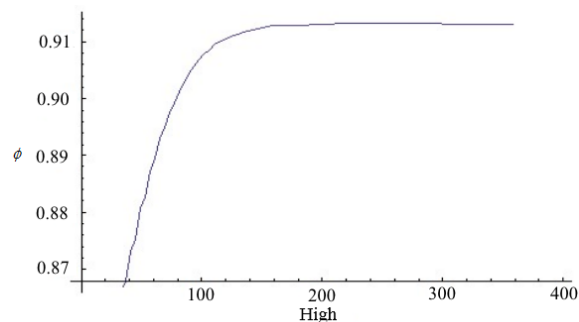


Рис. 3. Графік залежності ϕ від верхньої границі High з оператором Собеля

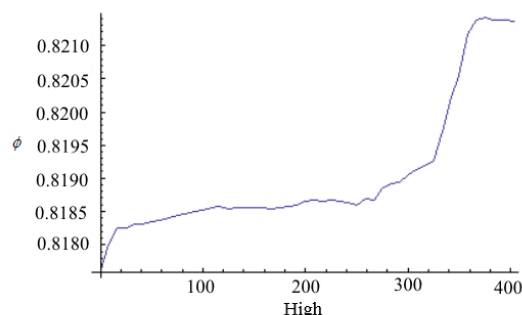


Рис. 4. Графік залежності ϕ від верхньої границі High з оператором Робертса

6. Висновки

В результаті проведених досліджень було виконано детальний опис алгоритму Кенні в розрізі застосування операторів згортки та зазначено їх вплив на ефективність локалізації об'єктів на зображенні.

Окрім цього запропоновано в якості критерію для вибору оптимального ядра згортки використовувати ймовірність того, що знайдена границя об'єкту на зображенні буде відповідати реальній, за рахунок обчислення відношення кількості вдало знайдених пікселів границь до загальної потужності множини пікселів на зображенні. Застосування оператору згортки в алгоритмі Кенні, що враховує його специфіку. За рахунок проведеного експериментального дослідження було доведено, що використання інших методів для пошуку градієнту, окрім оператора Собеля, вигідно тільки в окремих випадках. На практиці найбільш оптимальним є застосування оператору Собеля.

Проведене дослідження дозволило зробити акцент на відкриті проблеми даного алгоритму:

- після першого кроку алгоритму, згладжування, є ризик втратити чіткі границі;
- окрім цього, типовий прийом переходу до відтінків сірого також призводить до втрати інформації: близькі кольори зливаються в один;
- знайдені границі орієнтуються тільки у 4-х напрямках, а отже контури не змінюються плавно;
- також відкритою проблемою залишається чисельна оцінка того, наскільки якісно було знайдено границі.

Наведений план експериментального дослідження дозволяє оцінювати ефективність виділення границь та використовувати отримані результати у подальших наукових дослідженнях.

Література

1. Форсайт, Д. Компьютерное зрение. Современный подход [Текст] / Д. Форсайт, Ж. Понс. – М. : Вильямс, 2004. – 928 с.
2. Szeliski, R. Computer Vision: Algorithms and Applications [Text] / R. Szeliski. – London: Springer, 2011. – 812 p. doi: 10.1007/978-1-84882-935-0
3. Білінський, Й. Методи обробки зображень в комп'ютеризованих оптико-електронних системах [Текст] / Й. Білінський. – Вінниця: ВНТУ. – 2010. – 272 с.
4. Li, J. A Researchon Improved Canny Edge Detection Algorithm [Text] / J. Li, S. Ding // Communications in Computer and Information Science. – 2011. – Vol. 228. – P. 102–108. doi : 10.1007/978-3-642-23223-7_13
5. Harris, C. A Combined Cornerand Edge Detector [Text] / C. Harris, M. Stephens // Proceedings of the Alvey Vision Conference. – Plessey: Plessey Research Roke Manor, 1988. – P. 23.1–23.6. doi : 10.5244/c.2.23
6. Hines, E. Application of edge detection techniques to detection of the bright band in radar data [Text] / E. Hines // Image and Vision Computing. – 1983. – Vol. 1, Issue 4. – P. 221–226. doi : 10.1016/0262-8856(83)90021-5
7. Canny, J. F. A computational approach to edge detection [Text] / J. F. Canny // IEEE Trans Pattern Analysis and Machine Intelligence. – 1986. – Vol. PAMI-8, Issue 6. – P. 679–698. doi : 10.1109/tpami.1986.4767851
8. Conroy, L. Retrospective evaluation of visually monitored deep inspiration breath hold for breast cancer patients using edge detection [Text]/ L. Conroy, R. Yeung, S. Quirk, T. Phan, W. L. Smith // World Congress on Medical Physics and Biomedical Engineering. – Toronto, Canada, 2015. – P. 583–586. doi: 10.1007/978-3-319-19387-8_143
9. Harun, M. N. H. Comparative study of edge detection algorithm: vessel wall elasticity measurement for deep vein thrombosis diagnosis [Text]/ M. N. H. Harun, I. Nabilah, A. N. Shazilah // ARPN Journal of Engineering and Applied Sciences. – 2015. – Vol. 10, Issue 19. – P. 8635–8641. – Available at: http://www.arpnjournals.org/jeas/research_papers/rp_2015/jeas_1015_2772.pdf
10. Ramamurthy, B. Content based Image Retrieval for Medical Images using Canny Edge Detection Algorithm [Text]/ B. Ramamurthy, K. R. Chandran // International Journal of Computer Applications. – 2011. – Vol. 17, Issue 6. – P. 32–37. doi: 10.5120/2222-2831
11. Law, M. W. K. Weighted Local Variance-Based Edge Detection and Its Application to Vascular Segmentation in Magnetic Resonance Angiography [Text] / M. W. K. Law, A. C. S. Chung // IEEE Transactions on Medical Imaging. – 2007. – Vol. 26, Issue 9. – P. 1224–1241. doi: 10.1109/tmi.2007.903231
12. Hemalatha, R. Automated Driving Vehicle Using DigitalImage Processing [Text] / R. Hemalatha, E. Deepa, R. Sasipriya // IJISSET - International Journal of Innovative Science, Engineering and Technology. – 2015. – Vol. 2, Issue 9. – P. 445–446. – Available at : http://ijiset.com/vol2/v2s9/IJISSET_V2_19_55.pdf
13. Panetta, K. Logarithmic Edge Detection with Applications [Text] / K. Panetta, E. Wharton, S. S. Agaian // Journal of Computers. – 2008. – Vol 3, Issue 9. – P. 11–19. doi: 10.4304/jcp.3.9.11-19
14. Bradski, G. Learning OpenCV. [Text] / G. Bradski, A. Kaehler. – O'ReillyMedia, 2008. – 580 p. – Available at: <http://www.cse.iitk.ac.in/users/vision/dipakmj/papers/OReilly%20Learning%20OpenCV.pdf>
15. Labeled Faces in the Wild [Internet]. – Available at: <http://vis-www.cs.umass.edu/lfw/>
16. Steger, C. Machine Vision Algorithms and Applications [Text] / C. Steger, M. Ulrich, C. Wiedemann. – Wiley-VCH, 2008. – 360 p.
17. Johnson, S. On Digital Photography [Text] / S. Johnson. – O'Reilly Media, 2006. – 320 p.
18. Лисак, Н. В. Підвищення якості розпізнавання методом Віюлі-Джонса в задачах інформаційної безпеки підприємства шляхом попередньої обробки зображень [Текст] / Н. В. Лисак, Ю. В. Міронова, І. О. Марченко, С. О. Петров // Оптико-електронні інформаційно-енергетичні технології. – 2015. – Т. 29, № 1. – С. 70–75.