

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ РОЗПІЗНАВАННЯ ОБЛИЧ

Виконано аналіз ефективності застосування інформаційної технології розпізнавання облич людей із використанням одновимірних прихованих марківських моделей за допомогою створеного програмного продукту, розробленого на об'єктно-орієнтованій мові програмування Java у середовищі Eclipse.

Ключові слова: *прихована марківська модель, класифікація, алгоритм прямого-зворотного проходу, алгоритм Вітербі, алгоритм Баума–Велша.*

Выполнен анализ эффективности применения информационной технологии распознавания лиц людей с использованием одномерных марковских моделей с помощью созданного программного продукта, разработанного на объектно-ориентированном языке программирования Java в среде Eclipse.

Ключевые слова: *скрытая марковская модель, классификация, алгоритм прямого-обратного хода, алгоритм Витерби, алгоритм Баума–Велша.*

It is made the analysis of the of information technology effectiveness facial recognition of people using the one-dimensional Markov models. Analysis is based on using the created software product developed with object-oriented programming language Java within Eclipse.

Keywords: *hidden Markov model, classification, forward-backward algorithm, Viterbi algorithm, Baum–Welch algorithm.*

Вступ. Системи розпізнавання облич можуть бути використані для контролю доступу осіб до зон з обмеженим доступом, доступу до комп'ютерів, пошуку та розпізнавання осіб за спеціалізованими базами даних (наприклад, бази даних злочинців). Система розпізнавання облич має бути стійкою до умов, які змінюються (наприклад зміна освітлення, різні кути нахилу та повороту облич тощо).

Аналіз останніх досліджень і публікацій. Для розпізнавання облич найчастіше використовуються методи на основі головних компонент (власних облич) [1], лінійного дискримінантного аналізу [2], нейронних мереж [3; 4], прихованих марківських моделей [5; 6], метод еластичних графів [7].

Постановка задачі. Виконати аналіз ефективності застосування одновимірних прихованих марківських моделей для розпізнавання облич, провівши алгоритмізацію та створивши програмне забезпечення.

Основний матеріал. Прихована марківська модель складається з таких елементів: кінцевої множини прихованих станів системи $S = \{s_1, s_2, \dots, s_N\}$; кінцевої множини спостережуваних станів: $V = \{v_1, v_2, \dots, v_M\}$; матриці перехідних імовірностей $A = \{a_{ij}\}$, де $a_{ij} = P(q_t = s_j | q_{t-1} = s_i)$, $i, j = \overline{1, N}$, $t = \overline{1, T}$, q_t – стан моделі у момент часу t ; послідовності $Q = q_1, q_2, \dots, q_T$ випадкових величин зі значеннями із S ; вектора початкових імовірностей станів системи: $\pi = \{\pi_i\}$; матриці ймовірностей спостережуваних символів $B = \{b_i(k)\}$, $b_i(k) = P(o_t = v_k | q_t = s_i)$, $i = \overline{1, N}$, $k = \overline{1, M}$. o_t – спостережуваний символ у момент часу $t = \overline{1, T}$.

Кожній послідовності прихованих станів s_{i_1}, s_{i_2}, \dots ставиться у відповідність послідовність випадкових величин o_1, o_2, \dots зі значеннями з V . При цьому ймовірність спостереження деякого символу залежить тільки від того, в якому стані знаходиться система.

Навчання системи проводиться на зображеннях облич різних людей, кожній людині ставиться у відповідність одна прихована марківська модель, яка буде використана під час класифікації нових зображень. Етап навчання системи полягає в налаштуванні параметрів прихованої марківської моделі для максимізації ймовірності розпізнавання відповідної людини на основі максимізації функції $P(O^{(r)} | \lambda^{(r)})$ та $P(O^{(r)}, Q | \lambda^{(r)})$, де $\lambda^{(r)}$ – модель, яка відповідає r -й людині, $O^{(r)}$ – множина зображень однієї людини, Q – послідовність прихованих станів системи. Визначити послідовність прихованих станів дозволяє алгоритм динамічного програмування Вітербі. Для максимізації ймовірності спостереження послідовності $O^{(r)}$ за використанням відповідної прихованої марківської моделі $\lambda^{(r)}$ використовується алгоритм Баума – Велша. Після навчання системи $\lambda^{(r)}$ використовується для класифікації нових зображень. Навчання системи складається із двох етапів (рис. 1 та 2 відповідно [8]).

Початкові значення параметрів моделі можна вибирати довільно, ураховуючи умови нормування. Доведено, що алгоритм завжди збігається, при цьому майже завжди до точки локального максимуму.

Алгоритм динамічного програмування Вітербі використовується для визначення максимізуючої послідовності прихованих станів, при фіксованій послідовності спостережуваних символів.

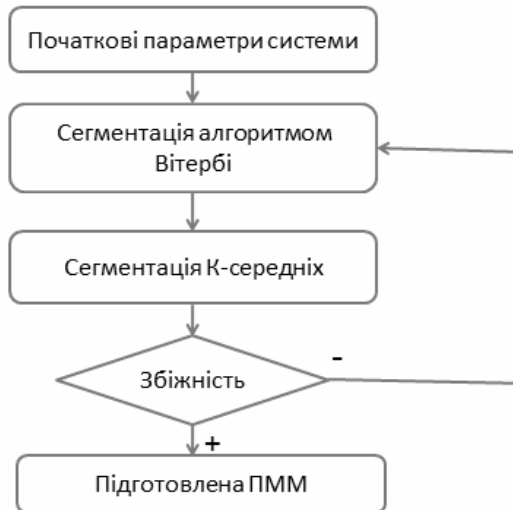


Рис.1. Початкова сегментація зображення

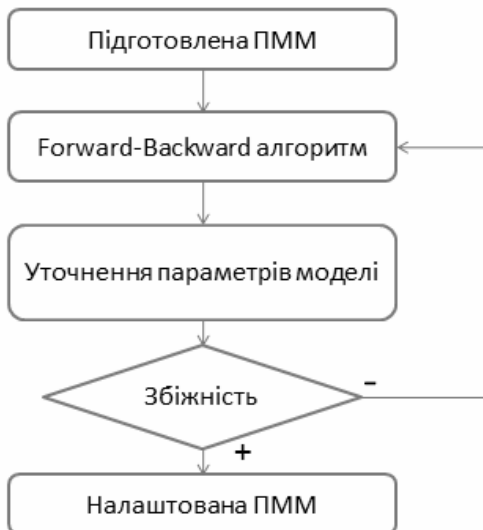


Рис.2. Уточнення параметрів моделі

Сегментація зображення за допомогою алгоритму К-середніх відбувається в межах кожного стану окремо, в якості початкового наближення алгоритму об'єкти рівномірно розподіляють за кластерами, для розрахунку відстані між об'єктами використовується метрика простору $L_2: \|x\|^2 = \sum_{i=1}^{\dim(x)} x_i^2$.

Згідно з [9] для ефективного обчислення ймовірності появи послідовності символів без будь-яких умов використовується forward-backward-алгоритм [10], оскільки розрахункова формула

$$P(O|\lambda) = \sum_{q_1, q_2, \dots, q_T} \left(\pi_{q_1} \prod_{t=1}^{T-1} a_{q_t, q_{t+1}} \prod_{t=1}^T b_{q_t}(o_t) \right)$$

має високу обчислювальну складність, а саме $2TN^T$. Forward-backward-алгоритм дозволяє провести розрахунок $P(O|\lambda)$ за $3TN^2$ обчислень.

Уточнення параметрів моделі відбувається таким чином:

$$\begin{aligned} \pi_i^* &= \frac{1}{K} \sum_{k=1}^K \gamma_1^{(k)}(i), \\ a_{ij}^* &= \frac{\sum_{k=1}^K \sum_{t=1}^{T^k-1} \xi_t^{(k)}(i, j)}{\sum_{k=1}^K \sum_{t=1}^{T^k} \gamma_t^{(k)}(i)}, \\ c_{im}^* &= \frac{\sum_{k=1}^K \sum_{t=1}^{T^k} \gamma_t^{(k)}(i, m)}{\sum_{k=1}^K \sum_{t=1}^{T^k} \sum_{\tilde{m}}^{M_i} \gamma_t^{(k)}(i, \tilde{m})}, \\ \mu_{im}^* &= \frac{\sum_{k=1}^K \sum_{t=1}^{T^k} \gamma_t^{(k)}(i, m) o_t^{(k)}}{\sum_{k=1}^K \sum_{t=1}^{T^k} \gamma_t^{(k)}(i, m)}, \\ \Sigma_{im}^* &= \frac{\sum_{k=1}^K \sum_{t=1}^{T^k} \gamma_t^{(k)}(i, m) (o_t^{(k)} - \mu_{im}^*)(o_t^{(k)} - \mu_{im}^*)^T}{\sum_{k=1}^K \sum_{t=1}^{T^k} \gamma_t^{(k)}(i, m)}. \end{aligned}$$

$$\text{Де } \gamma_t(i) = P(q_t = s_q | O, \lambda) = \frac{a_t(i) \beta_t(i)}{\sum_{j=1}^N a_t(j) \beta_t(j)}, \quad i = \overline{1, N}, \quad t = \overline{1, T},$$

$$\xi_t(i, j) = P(q_t = s_i, q_{t+1} = s_j | O, \lambda) = \frac{a_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{k=1}^N \sum_{p=1}^N a_t(k) a_{kp} b_p(o_{t+1}) \beta_{t+1}(p)},$$

$$i, j = \overline{1, N}, \quad t = \overline{1, T-1},$$

$$\gamma_t(i, m) = \gamma_t(i) \left(\frac{c_{im} N(a_t, \mu_{im}, \Sigma_{im})}{\sum_{h=1}^{M_i} c_{ih} N(a_t, \mu_{ih}, \Sigma_{ih})} \right)$$

Для зменшення розмірності простору символів використовується двовимірне дискретне косинус-перетворення, яке переводить зображення до частотної області.

Створена інформаційна технологія реалізована у вигляді програмного продукту, розробленого на об'єктно-орієнтованій мові програмування Java у середовищі Eclipse.

Практична апробація виконана на базі зображень Olivetty Research Ltd (ORL), яка складається із зображень 40 людей, по 10 зображень на людину. База містить напівтони нормалізовані зображення розміром 112x92 пікселів.

Тестування проводилися з використанням таких параметрів: розмір вікна сканування 10x10 пікселів; перекриття вікон сканування 7x7 пікселів; 16 коефіцієнтів дискретного косинус-перетворення.

Під час тестування програмного продукту отримані такі результати.

1. Досить висока якість розпізнавання (табл.1).

Таблиця 1

Залежність якості розпізнавання від розміру навчальної вибірки

Кількість зображень на персону – для навчання	Кількість зображень на персону – для екзамену	Час навчання, с	Час розпізнавання, с	Якість, %
1	9	861	4724	86
2	8	1494	4121	92
3	7	2459	3642	92
4	6	3815	3102	94
5	5	4542	2590	94

2. Використання діагональної дисперсійно-коваріаційної матриці дозволяє суттєво збільшити швидкість роботи системи, з незначним погіршенням якості розпізнавання (табл.2).

Таблиця 2

Залежність якості розпізнавання від розміру навчальної вибірки з використанням діагональної дисперсійно-коваріаційної матриці

Кількість зображень на персону – для навчання	Кількість зображень на персону – для екзамену	Час навчання, с	Час розпізнавання, с	Якість, %
1	9	133	520	86
2	8	225	480	90
3	7	327	401	93
4	6	398	346	92
5	5	525	290	91

3. Вдалий вибір параметрів впливає на швидкість та якість роботи системи (табл.3). У нашому випадку це: розмір вікна сканування 16x16 пікселів; перекриття вікон сканування 8x8

пікселів; використання 16 коефіцієнтів дискретного косинус-перетворення.

Таблиця 3

Залежність якості розпізнавання від розміру навчальної вибірки з використанням діагональної матриці

Кількість зображень на персону – для навчання	Час навчання, с	Час розпізнавання, с	Якість, %
1	32	77	80
2	40	70	92
3	53	60	94
4	61	51	95
5	78	43	95

4. Можливі два варіанти налаштування системи: для збільшення швидкодії системи з незначним погіршенням якості розпізнавання та для максимізації ймовірності розпізнавання зі значним програшем у швидкодії (табл.4).

Таблиця 4

Залежність якості розпізнавання від параметрів вікна сканування

Розмір вікна сканування	Перекриття	Кількість коефіцієнтів 2DDCT	Час навчання, с	Час розпізнавання, с	Якість, %
12×12	8×8	3×3	48	175	82
12×12	8×8	4×4	84	294	85
12×12	8×8	5×5	138	494	85
12×12	10×10	3×3	205	699	83
12×12	10×10	4×4	340	1189	86
12×12	10×10	5×5	340	1634	86
16×16	8×8	3×3	19	45	75
16×16	8×8	4×4	36	76	80
16×16	8×8	5×5	54	125	77
16×16	10×10	3×3	31	78	80
16×16	10×10	4×4	49	135	81
16×16	10×10	5×5	81	219	82

Висновки. Розроблена інформаційна технологія ідентифікації облич людини з використанням одновимірних марківських моделей дозволяє, із гарно налаштованими параметрами, досить швидко та

якісно розв'язувати задачу класифікації облич та може бути пристосована, залежно від необхідних умов, для збільшення швидкодії або якості розпізнавання.

Бібліографічні посилання

1. **Muller N.** Singular Value Decomposition, Eigenfaces, and 3D Reconstructions / N. Muller, L. Magaia, B.M. Herbst // SIAM Review. – 2004. – Vol. 46, No 3. – P. 518 – 545.
2. **Kim T.-K.** Component-based LDA Face Descriptor for Image Retrieval / Tae-Kyun Kim, Hyunwoo Kim, Wonjun Hwang // British Machine Vision Conference(BMVC), Cardiff, UK, 2002.
3. **Yahagi T.** Face Recognition using neural networks with multiple combinations of categories / T.Yahagi, H.Takano // International Journal of Electronics Information and Communication Engineering. – 1994. – Vol. J77-D-II, No.11. – P. 2151–2159.
4. **Latha P.** Face Recognition using Neural Networks / P.Latha, Dr.L.Ganesan, S.Annadurai // Signal Processing: An International Journal. – 2009. – Vol. 3. – Issue 5. – P. 153–160.
5. **Bicego M.** Using hidden Markov models and wavelets for face recognition / M. Bicego, U. Castellani, V. Murino // Image Analysis and Processing. – 2003. – P. 52–56.
6. **Nefian A.V.** Hidden Markov Models For Face Recognition / A.V. Nefian, M.H. Hayes // ICA SSP98. – 1998. – Vol. 5. – P. 2721–2724.
7. **Wiskott L.** Face Recognition by Elastic Bunch Graph Matching . Wiskott L., Fellous J.M., Kruger N. // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 1997. – 19(7). – P. 775–779.
8. **Kinscher J.** The Munster Tagging Project Mathematical Background / J.Kinscher, H.Trebbe // Arbeitsbereich Linguistik University of Münster, 1995.
9. **Rabiner L. R.** A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition / Proceedings of the IEEE. – 1989. – Vol. 77, No. 2. – P. 257–285.
10. **Dawei Shen** Some Mathematics for HMM / October 13th, 2008.

Надійшла до редколегії 10.07.2013 р.